



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

Implementación de dos robots mini sumos utilizando sistemas de radio control "SABERTOOTH" y dispositivos de comunicación inalámbrica "BLUETOOTH"

AUTORES:

Robles Idrovo, Gabriel Augusto

Vaca Barberán, Gabriel Enrique

Trabajo de Titulación previo a la obtención del grado de

INGENIERO EN TELECOMUNICACIONES

TUTOR:

Palacios Meléndez, Edwin Fernando

Guayaquil, Ecuador

13 de marzo del 2017



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por los
Sres. Robles Idrovo, Gabriel Augusto y Vaca Barberán, Gabriel Enrique
como requerimiento para la obtención del título de **INGENIERO EN
TELECOMUNICACIONES.**

TUTOR

Palacios Meléndez, Edwin Fernando

DIRECTOR DE CARRERA

Heras Sánchez, Miguel Armando

Guayaquil, a los 13 del mes de marzo del año 2017



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

DECLARACIÓN DE RESPONSABILIDAD

Nosotros, **Robles Idrovo, Gabriel Augusto y Vaca Barberán, Gabriel Enrique.**

DECLARAMOS QUE:

El trabajo de titulación **“Implementación de dos robots mini sumos utilizando sistemas de radio control “SABERTOOTH” y dispositivos de comunicación inalámbrica “BLUETOOTH”**” previo a la obtención del Título de **Ingeniero en Telecomunicaciones**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de nuestra total autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 13 del mes de marzo del año 2017

LOS AUTORES

ROBLES IDROVO, GABRIEL A.

VACA BARBERAN, GABRIEL E.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

AUTORIZACIÓN

Nosotros, **Robles Idrovo, Gabriel Augusto** y **Vaca Barberán, Gabriel Enrique**.

Autorizamos a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: **“Implementación de dos robots mini sumos utilizando sistemas de radio control “SABERTOOTH” y dispositivos de comunicación inalámbrica “BLUETOOTH”**”, cuyo contenido, ideas y criterios son de nuestra exclusiva responsabilidad y total autoría.

Guayaquil, a los 13 del mes de marzo del año 2017

LOS AUTORES

ROBLES IDROVO, GABRIEL A.

VACA BARBERAN, GABRIEL E.

REPORTE DE URKUND

URKUND

Documento [tesis final ROBLES - VACA 1.docx](#) (D26237761)

Presentado 2017-03-07 19:57 (-05:00)

Presentado por gabriel_robles15@outlook.com

Recibido edwin.palacios.ucsg@analysis.orkund.com

Mensaje RV: TESIS FINAL ROBLES - VACA [Mostrar el mensaje completo](#)

4% de esta aprox. 26 páginas de documentos largos se componen de texto presente en 7 fuentes.

Lista de fuentes	Bloques
Categoría	Enlace/nombre de archivo
⊕	TESIS-PARTE2.docx
⊕ >	cinthia tesis.docx
⊕	Tesis Rubén Parra.docx
⊕	Briones y Muñoz_FINAL_2017.docx
⊕	http://repositorio.ucsg.edu.ec/bitstream/33...

Reiniciar Exportar Compartir

0 Advertencia



UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA

DE INGENIERÍA

EN TELECOMUNICACIONES

TEMA:

Implementación

de dos robots mini sumos utilizando sistemas de radio control "SABERTOOTH" y dispositivos de comunicación inalámbrica "BLUETOOTH"

AUTORES: Robles IdrovoParra Sarango, Gabriel Augusto Vaca Barberáan, Gabriel EnriqueRubén

Danilo

DEDICATORIA

A nuestros padres que siempre estuvieron pendientes en toda nuestra carrera universitaria y nos ayudaron en lo que necesitábamos.

A nuestras abuelas Aurora y Ángela les dedicamos este triunfo tan anhelado por nosotros y que ellas tanto esperaban.

A nuestros queridos amigos que siempre dieron sus mejores deseos para que culminemos la carrera.

LOS AUTORES

ROBLES IDROVO, GABRIEL AUGUSTO
VACA BARBERAN, GABRIEL ENRIQUE

AGRADECIMIENTO

Agradezco infinitamente a Dios y a mis padres Wenceslao Enrique Vaca Sánchez y Myriam Lorena Barberán Soriano, por ser mi pilar fundamental en mi vida y siempre darme su apoyo incondicional durante todo el proceso de mis estudios.

Agradezco a María Elissa Coronel Zamora por siempre darme su apoyo desde el inicio de mi carrera universitaria hasta su culminación.

Agradezco a mi tutor M. Sc. Fernando Palacios por brindarme parte de sus conocimientos y por ser aparte de una gran persona un gran amigo.

Agradezco a mis amigos que siempre estuvieron ayudándome y poniendo su grano de arena para la obtención del título.

VACA BARBERÁN, GABRIEL ENRIQUE.

Antes que nada, agradezco a Dios por darme la fuerza y la salud para poder culminar mi carrera ya que es algo que anhelaba desde pequeño.

Agradezco a mis padres Cesar Augusto Robles Cueva Y Patricia de Fátima Idrovo Asqui por saber guiarme, apoyarme y servir como pilar fundamental para la obtención de este título.

Agradezco a mi tutor M. Sc. Fernando Palacios por brindarme parte de sus conocimientos y por ser aparte de una gran persona un gran amigo.

Agradezco a toda mi familia que de una manera u otra siempre me brindaron su apoyo cuando más lo necesitaba.

Agradezco a mis amigos que siempre supieron brindarme su ayuda de una u otra manera.

ROBLES IDROVO, GABRIEL AUGUSTO.



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TRIBUNAL DE SUSTENTACIÓN

f. _____

PALACIOS MELÉNDEZ, EDWIN FERNANDO
TUTOR

f. _____

HERAS SANCHEZ, MIGUEL ARMANDO
DIRECTOR DE CARRERA

f. _____

ZAMORA CEDEÑO, NÉSTOR ARMANDO
COORDINADOR DE ÁREA DE TELECOMUNICACIONES

Índice General

Índice de Figuras	XI
Índice de Tablas.....	XIII
Resumen	XIV
CAPÍTULO 1: Generalidades del Trabajo de Titulación.....	2
1.1. Introducción.....	2
1.2. Antecedentes.	3
1.3. Definición del Problema.....	5
1.4. Justificación del Problema.....	6
1.5. Objetivos del Problema de Investigación.....	6
1.5.1. Objetivo General.....	6
1.5.2. Objetivos Específicos.	6
1.6. Hipótesis.....	7
1.7. Metodología de Investigación.....	7
CAPÍTULO 2: Fundamentación Teórica.	8
2.1. Introducción a los microcontroladores.....	8
2.2. Evolución de los sistemas microcontroladores.....	9
2.3. Dispositivos microcontroladores PIC.....	11
2.3.1. Familia serie 12	12
2.3.2. Familia serie 16	14
2.3.3. Familia serie 18	17
2.4. Dispositivos microcontroladores Atmel.....	20
2.4.1. Atmega 164	21
2.4.2. Atmega 8	22
2.4.3. Atmega AT32UC3A	24
2.5. Sistema embebido Arduino	27
2.5.1. Arduino UNO	28

2.5.2. Arduino Micro	29
2.5.3. Arduino Nano	30
2.6. Comunicaciones por radiofrecuencia	31
CAPÍTULO 3: Diseño e Implementación.....	34
3.1. Principio de funcionamiento	34
3.2. Hardware.....	35
3.2.1. Sensor OMRON MR45 difuso	35
3.2.2. Sensores de reflexión IR	36
3.2.3. Motores fingertech robotics spark gearmotor	38
3.2.4. Placa Arduino Pro Micro.....	39
3.2.5. Módulo de comunicación Bluetooth HC-05	42
3.2.6. Controlador de motores de alto rendimiento Sabertooth 2x5.....	43
3.2.7. Baterías LiPo	44
3.2.8. Diseño electrónico	45
3.2.9. Diseño estructural.....	50
3.3. Software Implementado	52
3.3.1. Creación de aplicación SUMOAPP	52
3.3.2 Código fuente de prototipo robot minisumo autónomo y RC	59
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.....	65
4.1. Conclusiones.....	65
4.2. Recomendaciones.....	65
Bibliografía.....	66

Índice de Figuras

Capítulo 2

Figura 2. 1: Diagrama de bloques del PIC12F508/509	13
Figura 2. 2: Diagrama de bloques del PIC 16F8A.....	16
Figura 2. 3: Diagrama de bloques del PIC 18F2X2.....	19
Figura 2. 4: Diagrama de bloques de Atmega 164.....	22
Figura 2. 5: Diagrama de bloques del Atmega 8L.....	23
Figura 2. 6: Diagrama de bloques del Atmega AT32UC3A.....	25
Figura 2. 7: Diferentes placas Arduino UNO	28
Figura 2. 8: Placa del Arduino Micro frontal/posterior.	29
Figura 2. 9: Esquema de tarjeta Arduino Nano	31
Figura 2. 10: Principales componentes para implementar un sistema de comunicación RF.	32

Capítulo 3

Figura 3. 1: Sensor Omron MR 45.....	36
Figura 3. 2: Sensores Pololu QRT-1RC.....	37
Figura 3. 3: Esquemático del sensor Pololu QTR-1RC.....	37
Figura 3. 4: Motor DC Spark con caja reductora.....	38
Figura 3. 5: Curva característica de los motores Spark Gearmotor	39
Figura 3. 6: Pines de microcontrolador ATmega32u4.....	40
Figura 3. 7: Pines de Arduino Pro Micro	41
Figura 3. 8: IDE Arduino	41
Figura 3. 9: Módulo bluetooth HC-05.....	42
Figura 3. 10: Sabertooth 2x5	44
Figura 3. 11: Batería LiPo	45
Figura 3. 12: Etapa de alimentación	46
Figura 3. 13: Etapa de obtención de datos por sensores y serial	47

Figura 3. 14: Etapa de control de motores	48
Figura 3. 15: Diseño esquemático del circuito.	48
Figura 3. 16: Prueba de simulación.	49
Figura 3. 17: Diseño de PCB del robot minisumo	50
Figura 3. 18: Diseño estructural del minisumo	51
Figura 3. 19: Aplicativo online App Inventor.....	53
Figura 3. 20: Administración de proyectos de App Inventor.....	53
Figura 3. 21: Pantalla de herramientas para construcción de app.	54
Figura 3. 22: Pantalla principal de SUMOAPP.....	55
Figura 3. 23: Pantalla de opción autónomo de SUMOAPP.....	55
Figura 3. 24: Pantalla de mando RC de SUMOAPP.	56
Figura 3. 25: Código de bloques de pantalla principal.	57
Figura 3. 26: Código de bloques de pantalla autónomo.....	57
Figura 3. 26: Código de bloques de pantalla autónomo.....	58
Figura 3. 28: Cabecera de código fuente en Arduino.....	60
Figura 3. 29: Void setup de código fuente Arduino	61
Figura 3. 30: Funciones de movimiento.	61
Figura 3. 31: Función que recibirá valores del módulo bluetooth y evaluará.	62
Figura 3. 32: Función principal de código fuente.	64

Índice de Tablas

Capítulo 2

Tabla 2. 1: Algunos miembros de la familia serie 16..... 14

Tabla 2. 2: Algunos miembros de la familia serie 18..... 18

Capítulo 3

Tabla 3. 1: Características principales del módulo Bluetooth HC-05 43

Resumen

Podemos decir que nuestro trabajo está fundamentado en las transmisiones inalámbricas y la microelectrónica. Usamos microcontroladores los cuales están compuestos por dispositivos electrónicos que ayudan al funcionamiento del microcontrolador. Todo microcontrolador necesita un software que le ordene realizar una función específica. El enfrentamiento de robots minisumo es una batalla en la cual luchan 2 robots hasta sacar al oponente del dohyo, estos robots pueden ser autónomos y no autónomos los cuales pueden ser controlados por wifi, bluetooth y radiofrecuencia. La radiofrecuencia se la utiliza para el espectro electromagnético. Bluetooth es una comunicación inalámbrica de corto alcance. Mientras, que WI-FI es la tecnología más usada para la comunicación de dispositivos electrónicos de corto y largo alcance dependiendo su capacidad. Los dos prototipos implementados cumplen reglamentaciones para las participaciones en los próximos torneos de robótica. Finalmente, los diseños fueron validados en diferentes competencias logrando ubicarse entre los primeros lugares, y obteniendo certificaciones para torneos en Colombia y México.

Palabras claves: MINISUMO, PIC, ATMEGA, ARDUINO, RADIOFRECUENCIA, BLUETOOTH.

CAPÍTULO 1: Generalidades del Trabajo de Titulación.

1.1. Introducción.

El trabajo de titulación está dirigido para personas con conocimientos de microelectrónica y transmisión inalámbricas. Ya se sabe que un microcontrolador (abreviado uC) es un circuito integrado (CI). Este CI internamente dispone de: (a) una Unidad Central de Procesamiento (*Central Processing Unit, CPU*), (b) memorias de acceso aleatorio (*Random Access Memory, RAM*) y de solo lectura (*Read Only Memory, ROM*), (c) pines de E/S y, (d) periféricos. Estos elementos son interconectados internamente en el uC, y generalmente, se la conoce 'microcomputadora'. Por lo tanto, se entiende que un uC es en sí, una microcomputadora completa encapsulada en un circuito integrado.

Todo uC necesita un software que le ordene realizar una función específica. Normalmente esto se guarda en la memoria ROM. Cabe recalcar que sin un programa determinado los uC carecen de utilidad. Las aplicaciones de los uC son varias, lo único que las limita es la imaginación de los usuarios. Comúnmente se encuentran uC dentro de áreas como la robótica y el automatismo, en las telecomunicaciones, en la industria automotriz, etc.

La lucha entre robots mini-sumos, son competencias inspiradas en la lucha japonesa, en este caso se enfrentan 2 robots con el objetivo de localizar, atacar y sacar fuera del campo de batalla (dohyo) a su oponente.

Los robots mini-sumos pueden ser controlados por medios de transmisión inalámbricos como: radiofrecuencia, bluetooth y wifi.

Radiofrecuencia: la radio frecuencia se utiliza para las frecuencias que se utilizan en las radiocomunicaciones con respecto al espectro electromagnético.

Bluetooth: bluetooth, es una tecnología personal inalámbrica (WPAN), también se la conoce como una red inalámbrica de corto alcance, que es utilizada para conectar dispositivos sin la necesidad de utilizar un cable. Con Bluetooth no es necesario que se visualice directamente a los dispositivos para comunicarse, esto hace que su uso sea más flexible.

WIFI: la comunicación wifi es una conexión de forma inalámbrica entre dispositivos electrónicos. Los dispositivos que generalmente usan esta tecnología wifi, (laptops, Smart tv, videojuegos, Smartphone, etc.) se pueden conectar a internet mediante wifi el cual es un punto de acceso inalámbrico. El alcance del wifi varía en interiores y exteriores dependiendo de la capacidad de routers en interiores suele ser 20 m, en exteriores suele ser mucho mayor.

1.2. Antecedentes.

Un emisor de radio control (RC) es una fuente inalámbrica que es posible conectarlo a un dispositivo Arduino el cual se lo puede emplear como controlador. Los emisores de (RC) son los que se encuentran fácilmente en el

campo de la robótica y la electrónica, estos se los emplea para manipular vehículos que utilicen control remoto.

En la actualidad contamos con una gran diversidad de emisores RC, los cuales varían su resolución, alcance y número de canales que contienen. La principal variable son los números de canales que contienen los emisores RC los cuales representan las variables analógicas que vamos a poder controlar en el receptor. Por ejemplo:

- En medios de movilización terrestres y barcos, los cuales se mueven de forma horizontal, vamos a necesitar un mínimo de 2 canales los cuales son (aceleración y giro).
- En medios de movilización aérea como aviones, helicópteros o quadricópteros se necesitarán mínimo de 4 canales (aceleración, y tres ángulos).

Por lo general, los transmisores básicos constan de 6 canales, esto es una inversión que no haremos muy a menudo por lo cual es aconsejable que no se adquieran transmisores con un número menor de canales. Por lo que los transmisores de 9 y 10 canales se pueden considerar como profesionales, pero su precio es muy elevado estos son destinados a vehículos más complejos.

En la actualidad la mayor parte de emisoras funcionan en una banda de 2.4 GHz, la cual es una frecuencia libre. La distancia de estas puede ser desde

unos metros hasta varios km. La resolución varía entre 10bits (1024 niveles) a 12bits (4096 niveles). Junto al emisor vamos a necesitar un receptor que sea compatible con el mando que se utilizara. Es común que cuando se adquiere el mando el receptor venga incluido con él, aunque se pueden adquirir más receptores para que sean integrados en múltiples proyectos reajustando el transmisor.

Las emisoras RC son uno de los controles más sofisticados que se pueden emplear en proyectos de electrónica, y también es uno de los que cuentan con el mayor alcance. En general, será uno de los mandos preferidos para vehículos y robots complejos como el mini-sumo.

1.3. Definición del Problema.

En las telecomunicaciones se ha logrado el crecimiento y desarrollo de equipos de radio control que pueden ser empleados por medio de diferentes tipos de comunicaciones inalámbricas como lo son el Bluetooth, wifi, etc. Por lo que es necesario saber los diferentes tipos de dispositivos que se emplean para el uso de estas tecnologías. Actualmente existen varios proyectos de robots mini-sumos autónomos, y muy pocos dirigidos por radiocontrol.

Por la tanto, surge la necesidad de realizar la implementación de robots mini sumo, uno con total autonomía y el otro radio controlado, esté último su enfoque es las comunicaciones inalámbricas para su correcto funcionamiento y manejo de los mismos.

1.4. Justificación del Problema.

Con la gran popularidad adquirida en el campo de la robótica y las telecomunicaciones por los robots mini-sumos autónomos y no autónomos, hemos decidido hacer una implementación la cual será enfocada en el radiocontrol utilizado por los robots. En este proyecto, también explicaremos el uso y funciones de los diferentes tipos de tecnologías empleadas para el radiocontrol de estos robots, con sus respectivos análisis y conclusiones.

1.5. Objetivos del Problema de Investigación.

1.5.1. Objetivo General.

Diseñar e implementar dos robots mini sumos utilizando sistemas de radio control y dispositivos de comunicación inalámbrica Bluetooth.

1.5.2. Objetivos Específicos.

- Describir los fundamentos teóricos de los sistemas embebidos Arduino y de los microcontroladores Atmega.
- Realizar el diseño electrónico del sistema de comunicación por radio frecuencia y Bluetooth.
- Diseñar las interfaces virtuales en App Inventor y LabView para el control del minisumo.
- Evaluar el funcionamiento del robot minisumo en competencias de robótica en representación de la Facultad de Educación Técnica para el Desarrollo.

1.6. Hipótesis.

La hipótesis planteada es que, una vez realizada la respectiva implementación, con los resultados obtenidos de cada señal de radiofrecuencia podamos ver las diferencias de cada una de estas y así determinar cuál es la de mayor calidad para ser utilizada en los robots mini-sumos.

1.7. Metodología de Investigación.

Nuestro trabajo de titulación se realizó mediante tres procesos de investigación, los cuales son: exploratoria, descriptiva y explicativa.

- Estudios exploratorios: realizamos estos estudios porque nuestro objetivo es examinar un tema muy poco visto por nosotros, el cual nos ayudara para la implementación de los mini-sumos.
- Estudios descriptivos: con este método de investigación nuestro propósito es descubrir las diferentes funciones de los robots mini-sumos, para así especificar las propiedades importantes de nuestro proyecto.
- Estudios explicativos: con este estudio podremos saber el comportamiento de una variable relacionada con otra, es decir los problemas que se presenten mediante la implementación de los mini-sumos.

CAPÍTULO 2: Fundamentación Teórica.

2.1. Introducción a los microcontroladores.

Un microcontrolador es un chip de computador, micro indica que es diminuto y controlador nos habla sobre el uso del mismo que es de controlar objetos, procesos o eventos. Otros términos que lo describen es de un control embebido, porque controla y da soporte a los circuitos que a menudo están contruidos en un dispositivo que ellos controlan.

Según M. Gualli y R. Véliz (2016). Un microcontrolador es un microcomputador integrado en un solo chip, internamente se estructura de una CPU, memoria interna para almacenar datos, memoria de programa no volátil.

Según lo dicho, un microcontrolador es un chip de computador que contiene memoria de interfaz de entrada y salida adicional a la del CPU, ya que la cantidad de memorias e interfaces ajustables en un chip es limitada, los microcontroladores suelen usar es más pequeña que la que requiere para el soporte de los componentes.

2.2. Evolución de los sistemas microcontroladores.

El desarrollo en el campo de los microcontroladores tuvo inicio en las tecnologías de circuitos integrados, esto nos ha permitido almacenar cientos de miles de transistores en un chip, eso era una condición previa para la fabricación de microprocesadores, se puede encontrar microcontroladores en todo tipo de cosas, cualquier dispositivo que mida, almacena, calcule, controle o muestra información es un candidato para implementar un microcontrolador dentro.

En el año 1969 un equipo de ingenieros japoneses (BUSICOM) llegó a los Estados Unidos pidiendo unos circuitos integrados para calculadoras que se diseñaran de acuerdo a sus proyectos. La solicitud fue acatada por Intel y Marcian Hoff estuvo al mando del proyecto, al tener experiencia trabajando con computadoras comenzó a dar soluciones distintas en lugar del diseño sugerido, en si suponía el funcionamiento del circuito integrado debería ser determinado para el programa almacenado en el circuito, requería de menos programación pero de más memoria que el proyecto ya propuesto, luego de que la idea de Marcial Hoff ganó tuvo la ayuda de Federico Faggin para convertir la idea en un producto luego de nueve meses, Intel logró desarrollar un producto de este tipo a partir del concepto original.

En 1971 Intel luego de obtener los derechos de BUSICOM sacó al mercado un microprocesador llamado 4004, un microprocesador de 4 bits con la velocidad de 6.000 operaciones por segundo.

En 1972 apareció en el mercado el microprocesador de 8 bits llamado 8008 gracias a Intel y Texas Instruments. Este microprocesador fue capaz de abarcar 16.000 de memoria, 45 instrucciones y 300.000 operaciones por segundo, este fue el procesador de todos los microprocesadores actuales.

En abril de 1974 Intel siguió desarrollando y lanzó el microprocesador de 8 bits llamado 8080 que fue capaz de almacenar 76 Kb de memoria y 75 instrucciones.

En enero de 1975 una popular revista de características electrónicas describe la Altair 8800, la primera microcomputadora que los aficionados podían crear y programar ellos mismos.

La Altair 8800 no era un elemento esencial para una computadora personal, fue programado por un conmutador de palanca en la parte frontal y tenía 256 bytes de ram en su versión estándar.

La caída del uso de la Altair cuando Microsoft ofreciera una versión de lenguaje básico de programación para ello, gracias a eso las computadoras personales tienen drivers y megabytes de RAM.

En la exposición de WESCOM en los Estados Unidos MOS Technology lanzó al mercado los procesadores 6501 y 6502, estos microprocesadores de 8 bits con 56 instrucciones y capacidad para dirigir directamente 64kb de memoria, por su bajo precio se hicieron populares por lo que se instaló en distintos ordenadores de la época (Apple I, Apple II, Atari, Commodore, Bellota, Oric, Galeb, Oraq, Ultra y muchos otros).

En 1976 Zilog anunció el Z80, ya como se había desarrollado el 8080 y se dieron cuenta que muchos seguirían utilizándolo, este microprocesador

tenía que ser compatible con el 8080 además de muchas otras características se han agregado de modo que el Z80 era el microprocesador más potente en ese momento por lo que fue capaz de dirigir directamente 64kb de memoria, tenía 176 instrucciones, un gran número de registros, una opción integrada para refrescar la memoria RAM dinámica, una sola fuente de alimentación, una mayor velocidad de funcionamiento, etc.

En el mismo año Intel presento el 8085 una versión actualizada del microprocesador de 8 bits, sin embargo, el Z80 fue mucho mejor que Intel perdió la batalla. No hubo grandes mejoras que pudieran los fabricantes de cambiar de opinión, por lo que el 6502, 6800 y el Z80 siguieron siendo los principales microprocesadores de 8 bits de ese tiempo.

2.3. Dispositivos microcontroladores PIC.

Los dispositivos microcontroladores PIC son originarios de la compañía General Instruments, en sus inicios eran diseñados para ser un simple controlador de aplicaciones, por esa la razón de su nombre control de interfaz periférica, el CPU simple del mismo contaba con una estructura RISC, con un solo registro de trabajo y solo 30 instrucciones.

El alcance de estos microprocesadores se pueden encontrar literalmente cientos de diferentes dispositivos, incluidos en diferentes paquetes de aplicaciones, por lo cual las características que todo tiene en común son de bajo costo, de 8 bits, acumulador único RISC (el registro de trabajo o w), estructura Harvard, autosuficientes, vectores fijos de reinicio y de interrupción.

La clara evolución de una familia de PIC con otra, así tenemos que cada miembro de alguna familia comparte la misma arquitectura de núcleo y el mismo set de instrucciones, son identificadas dependiendo los dos primeros dígitos de cada código del dispositivo. La “C” en la numeración implica que contiene tecnología CMOS, la “F” en la numeración que contiene tecnología flash (todavía usa CMOS como tecnología de núcleo), una “A” después del número indica una actualización del primer dispositivo que se emitió, una “X” indica que cierto número de valores, el que se toma no es importante para el número total citado.

2.3.1. Familia serie 12

Según T. Wilmshurst. (2007), esta serie están diseñados para aplicaciones muy pequeñas, embalados en pequeños circuitos integrados, con arquitectura simple y a pesar de su tamaño llevan algunos periféricos incluyendo convertidores A/D y memoria de datos EEPROM.

Es la serie más utilizada para introducir el rango y uno de los más simples de los microprocesadores, sus características serán reconocibles en PICs más avanzados, el tamaño de palabras de instrucción es de 12 a 14 bits, puede recibir hasta 33 instrucciones y no cuentan con interruptor de vectores. Los elementos principales de esta serie de microcontroladores, el núcleo, memoria de programa, datos de memoria o RAM, ruta de datos y periféricos.

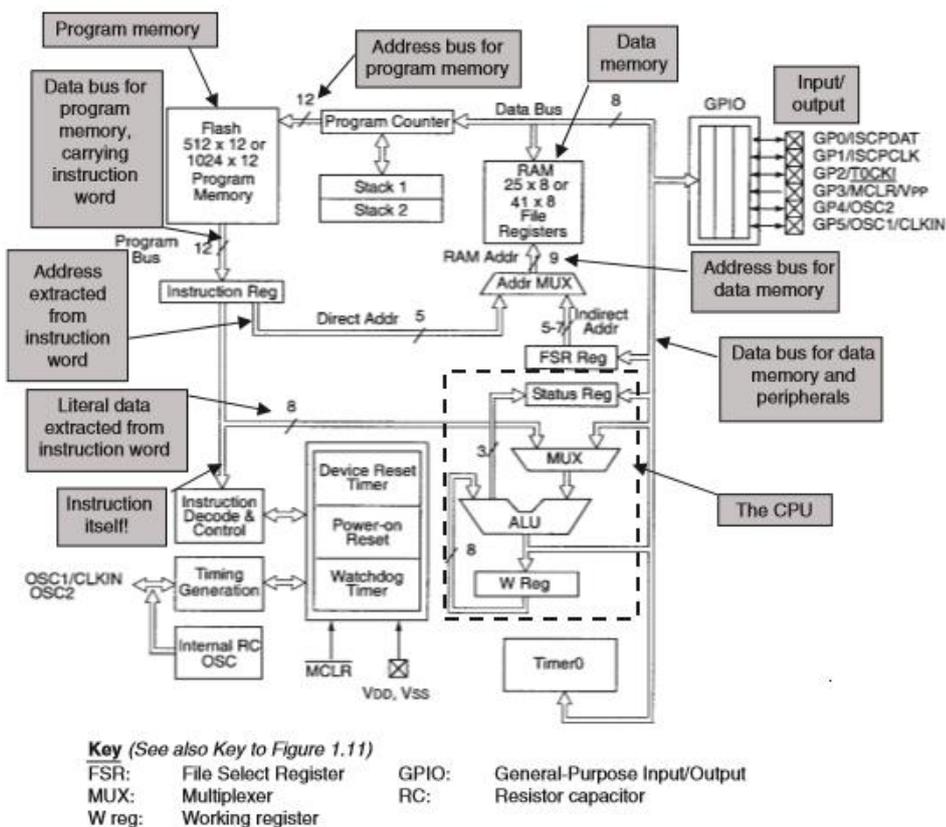


Figura 2. 1: Diagrama de bloques del PIC12F508/509
Fuente: (Wilmshurst. T., 2007).

El CPU es parte esencial del ALU (unidad lógica aritmética), el registro de trabajo y el estado del registro, este registro lleva un numero de bits que brindan información sobre el más reciente resultado de la instrucción realizada. Contiene entre 25 a 41 bytes de memoria de datos, arquitectura Harvard y el programa de bus de direcciones surge por el contador de programa y va solo a la memoria del programa, este es de 112 bits por lo que se puede dirigir a 4096 direcciones, la memoria de programa puede tener solo 512 en 1024 palabras.

Este Microcontrolador tiene solo dos dispositivos periféricos “on-chip”, un temporizador y el puerto de propósito general de entrada/salida, cuenta

con varias funciones relacionadas con el oscilador del reloj, la fuente de alimentación y reseteo.

2.3.2. Familia serie 16

Esta familia es muy extensa y diversa, pero permanece correcta en el concepto de familia extendida teniendo un núcleo y un set de instrucciones idénticas, pero con una diferencia de periféricos. Dentro de la familia extendida de la serie 16, se encuentran agrupaciones de controladores relacionados.

Tabla 2. 1: Algunos miembros de la familia serie 16

Numero de Dispositivo	Numero de pines	Velocidad del Reloj	Memoria (K= Kbytes i.e 1024 bytes)	Periféricos/Características Especiales
16F84A	18	DC DE 20 MHz	- 1K de memoria de programa - 68 bytes de RAM - 64 Bytes de EEPROM	1 Temporizador de 8 bit 1 Puerto paralelo de 5 bit 1 Puerto paralelo de 8 bit
16LF84A	Igual que su antecesor	Igual que su antecesor	Igual que su antecesor	Igual que su antecesor, pero con un rango de suministro de voltaje extendido
16F84A-04	Igual que su antecesor	DC de 4MHz	Igual que su antecesor	Igual que su antecesor
16F873A	28	DC de 20 MHz	4K de memoria de programa	3 puertos paralelos 3 contadores/temporizadores

			192 bytes de RAM 128 bytes de EEPROM	2 módulos de captura/comparación/PWM 2 módulos de comunicación en serie 5 canales ADC de 10 bits 2 comparadores analógicos
16F874A	40	DC de 20 MHz	4K de memoria de programa 192 bytes de RAM 128 bytes de EEPROM	5 puertos paralelos 3 contadores/temporizadores 2 módulos de captura/comparación/PWM 2 módulos de comunicación en serie 8 canales ADC de 10 bits 2 comparadores analógicos
16F876A	28	DC de 20 MHz	8K de memoria de programa 368 bytes de RAM 256 bytes de EEPROM	3 puertos paralelos 3 contadores/temporizadores 2 módulos de captura/comparación/PWM 2 módulos de comunicación en serie 5 canales ADC de 10 bits 2 comparadores analógicos
16F877A	40	DC de 20 MHz	8K de memoria de programa 368 bytes de RAM 256 bytes de EEPROM	5 puertos paralelos 3 contadores/temporizadores 2 módulos de captura/comparación/PWM 2 módulos de comunicación en serie 8 canales ADC de 10 bits 2 comparadores analógicos

Fuente: (Wilmshurst. T., 2007).

Muchos controladores hoy en día tienen programa de memoria on-chip, esta usualmente es EPROM (memoria programable de solo lectura borrable) con el asistente de ciclo de borrado EPROM de consumo de tiempo, otros controladores de la familia eligen usar EEPROM (memoria programable de solo lectura borrable eléctricamente) para el programa de memoria, así pueden programarse rápidamente y cambiar repetidamente.

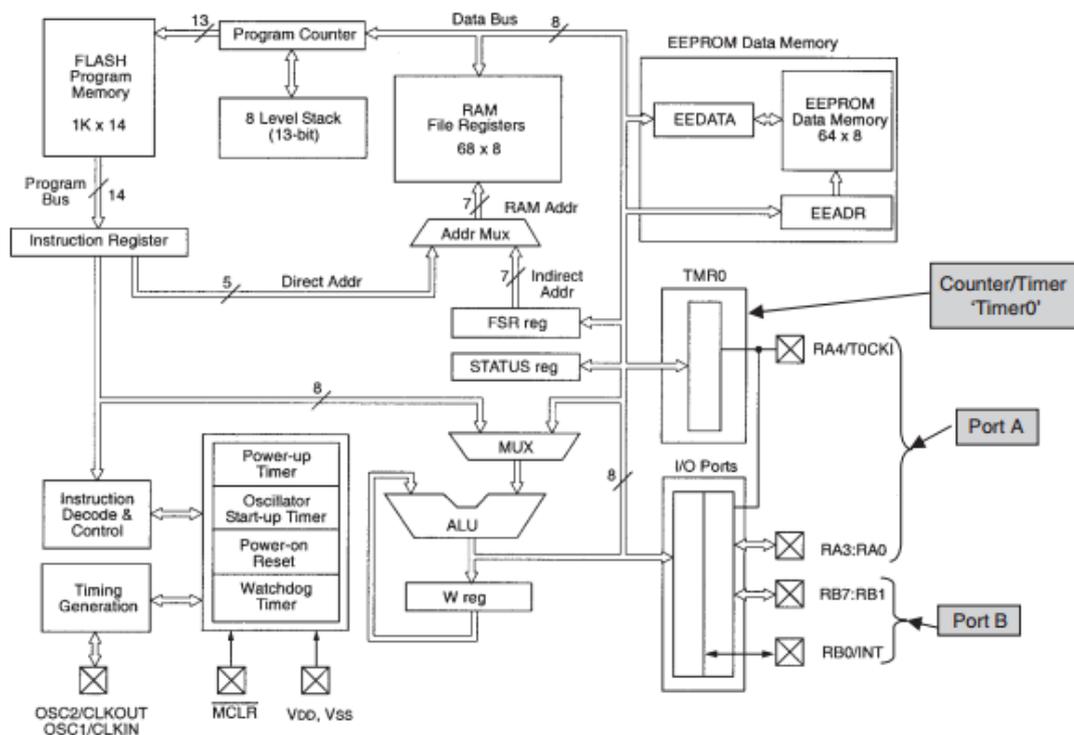


Figura 2. 2: Diagrama de bloques del PIC 16F8A.
Fuente: (Wilmshurst. T., 2007).

A comparación con el diagrama de bloques del PIC 12F508 se observan similitudes y diferencias, con 18 pines, no hay presión intensa para las distintas funciones en cada pin. El tamaño del bus de datos ha sido incrementado para llenar las necesidades de la familia serie 16, el programa de bus de direcciones es ahora de 13 bits y las palabras de instrucción de 14

bits por lo tanto tienen 8192 espacios de memoria, memoria de programa de 1k, el tamaño de la RAM se arrastra cuidadosamente en 68 localizaciones y la pila de 8 localizaciones.

Esta familia cuenta con 2 puertos digitales de entrada y salida, puerto A con 5 pines y el puerto B con 8, existe la adición de una capacidad de interrupción. En general esta familia provee una increíble diversidad y usabilidad en aplicaciones pequeñas.

2.3.3. Familia serie 18

Los microcontroladores PIC de la familia serie 18 mejoran el núcleo PIC, adecuándolo para proyectos integrados avanzados, con características nuevas pueden lograr la migración de un dispositivo de la serie 16 sea más fácil, en si fue diseñado para realizar el cambio con algunas características familiares.

Esta serie consta con un número de registro que se dobla con 16 bits de palabras de instrucción, un registro mejorado, interrupciones externas, dos vectores de interrupciones priorizadas, diferentes radicales aprovechando la estructura de la memoria y mayor tamaño en ella, generación de direcciones mejorada para la memoria de programa y datos, pila más grande con algún usuario de acceso/control y un bucle de bloqueo de fase generado por reloj.

Tabla 2. 2: Algunos miembros de la familia serie 18

Numero de Dispositivo	Numero de pines	Memoria	Periféricos/Características Especiales
18F242	28	16KB memoria de programa 8k instrucciones 768 bytes de RAM 256 bytes de EEPROM	3 puertos paralelos 4 contadores/timers 2 comparadores 2 módulos de comunicación serial 5 canales ADC de 10-bit
18F252	28	32KB memoria de programa 16k instrucciones 1536 bytes de RAM 256 bytes de EEPROM	3 puertos paralelos 4 contadores/timers 2 comparadores 2 módulos de comunicación serial 5 canales ADC de 10-bit
18F442	40	16KB memoria de programa 8k instrucciones 768 bytes de RAM 256 bytes de EEPROM	5 puertos paralelos 4 contadores/timers 2 comparadores 2 módulos de comunicación serial 8 canales ADC de 10-bit
18F452	40	32KB memoria de programa 16k instrucciones 1536 bytes de RAM 256 bytes de EEPROM	5 puertos paralelos 4 contadores/timers 2 comparadores 2 módulos de comunicación serial 8 canales ADC de 10-bit

Fuente: (Wilmshurst. T., 2007).

Todos estos dispositivos cuentan con 75 instrucciones, con oscilador de reloj DC de 40MHz, con versiones “low-power/bajo poder”. Los microcontroladores de esta familia cuentan de 28 pines, el CPU contiene 8 bit ALU, registro de trabajo “WREG” (a veces llamados acumulador) y un

hardware de 8 bit por 8 bit, registro de estado, la memoria del programa, el bus de direcciones entra a la memoria “Address Latch”. Con 21 bits es posible tener 2^{21} localizaciones e direcciones en 2 Mbytes.

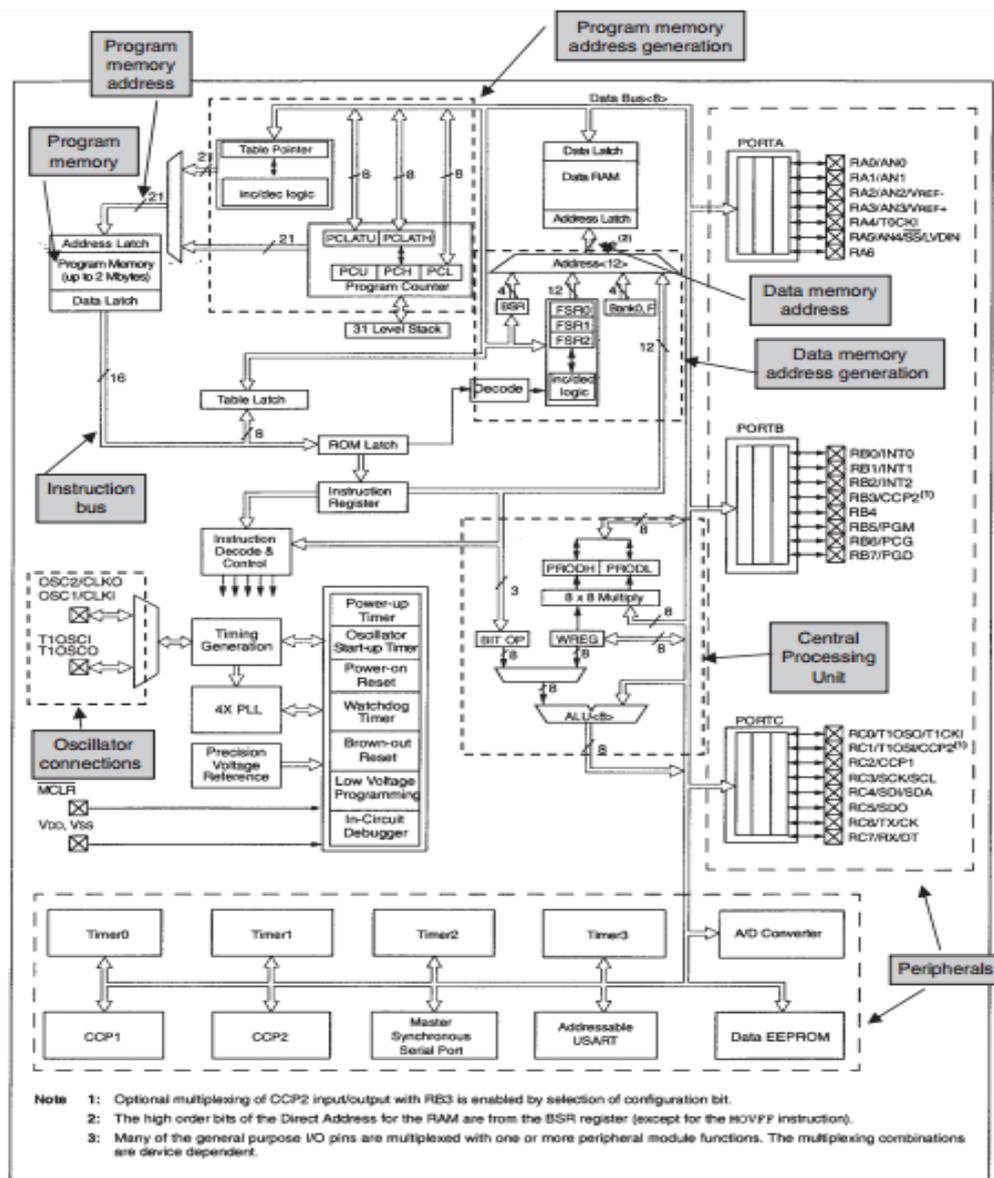


Figura 2. 3: Diagrama de bloques del PIC 18F2X2

Fuente:(Wilmshurst. T., 2007).

Se puede apreciar el bus de 16 bits que lleva las palabras de instrucción se ven dejando libre los datos de la memoria del programa que se encuentra en el área etiqueta, debajo del contador esta la pila que contiene 31

ubicaciones. Cuando el programa de memoria genera las direcciones se forma un bloque del diagrama en general con un banco de registro (FSR0) y un banco de selección de registros (BSR), la memoria de datos de direcciones es de 12 bits, que pueden direccionar 4096 bytes y la transferencia desde la memoria de datos se realiza a través del bus de datos principal.

El nuevo elemento del oscilador es el bucle de bloqueo de fase (PPL), esto introduce una opción de multiplicar la frecuencia del oscilador de modo que se acelera el funcionamiento en general, esto tiene impacto en la flexibilidad operativa y en el consumo de energía.

. La serie 18 añade 2 nuevos bits. Estos son OV (bit 3), que indica un sobre flujo del rango de 8 bits, y N, lo que indica que un número de complemento dos es negativo, estos bits extras nos permiten mejorar la ramificación del programa y una mejor capacidad matemática.

2.4. Dispositivos microcontroladores Atmel.

Según J. Beltrán (2016) estos dispositivos se diferencian de los otros microcontroladores de 8 bits por estar formados por 32 registros y un conjunto ortogonal de instrucciones. Según F. Barret (2009). Estos microcontroladores tienen una operación eficiente el programa, ya que las instrucciones más complejas están construidas sobre las operaciones básicas.

Según lo dicho, los microcontroladores Atmel constan con 32 registros de 8 bits para propósitos generales que están acopladas a la unidad lógica aritmética (ALU) dentro del CPU. Tienen una arquitectura Harvard y tienen memorias separadas dedicadas y buses para la información de programas y datos. Cuenta con un procesador RISC está equipado con complementos de operaciones básicas muy fáciles y eficientes. Las bases de registro de la arquitectura acoplada con el conjunto de instrucciones de RISC, permiten la ejecución de programas de manera más rápida y eficiente que permite completar al procesador la ejecución de instrucciones.

2.4.1. Atmega 164

El Atmel ATmega164 posee 131 tipos de instrucciones RISC. Puede ser ejecutada en un solo ciclo de reloj. El principal objetivo del procesador RISC es una simplificar el hardware y las instrucciones con el propósito de incrementar el rendimiento general de la computadora. El Atmega164 además está equipado con hardware adicional que permite la multiplicación en dos ciclos de reloj. En algunas otras arquitecturas de microcontroladores, la multiplicación requiere de manera típica más de dos ciclos de reloj.

Las características principales constan en 16k bytes de flash programable en el sistema con capacidad de lectura/escritura, EEPROM de 512 bytes, SRAM de 1k bytes, 32 líneas de entrada/salida de uso general, contador en tiempo real, tres contadores flexibles con comparación y PWM, dos USARTs programables en serie, un byte orientado a dos interfaces

seriales de alambre, una interfaz de 8 canales ADC de 10 bits con etapa de entrada diferencial opcional con ganancia programable.

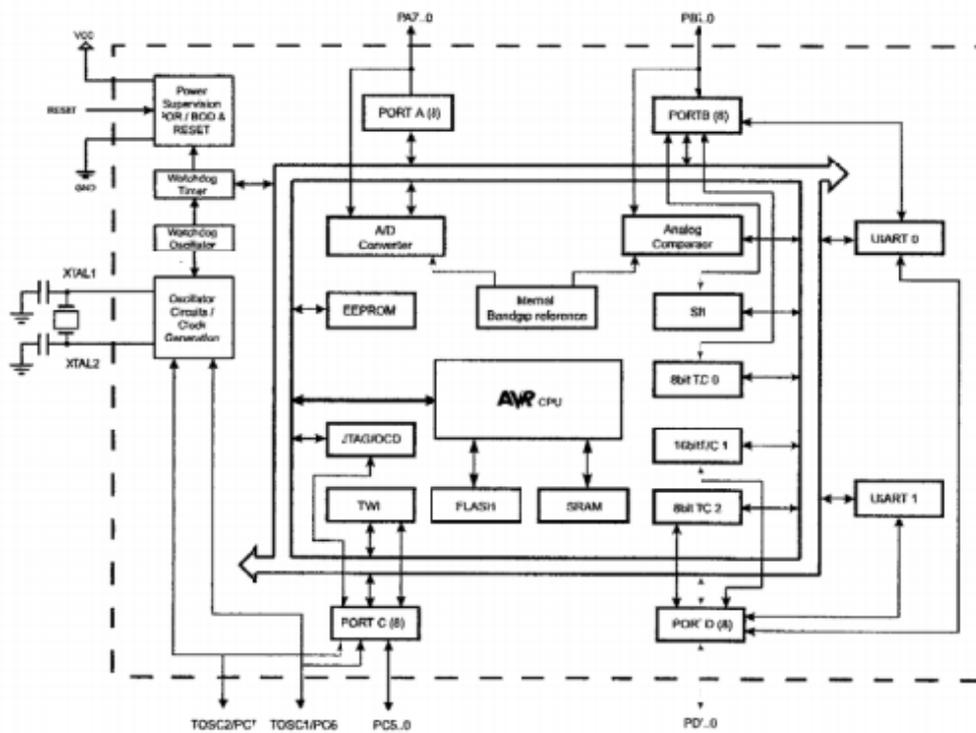


Figura 2. 4: Diagrama de bloques de Atmega 164
Fuente: (S. Barret, 2009).

2.4.2. Atmega 8

El Atmega 8 combina un gran set de instrucciones con 32 registros de trabajo para propósitos generales, están directamente conectadas con la unidad aritmética lógica ALU, permitiendo 2 registros independientes que dan acceso en una sola instrucción ejecutada en un ciclo de reloj. En la arquitectura que presentan obtienen códigos más eficientes ya que su

rendimiento es 10 veces más rápido que los microcontroladores convencionales.

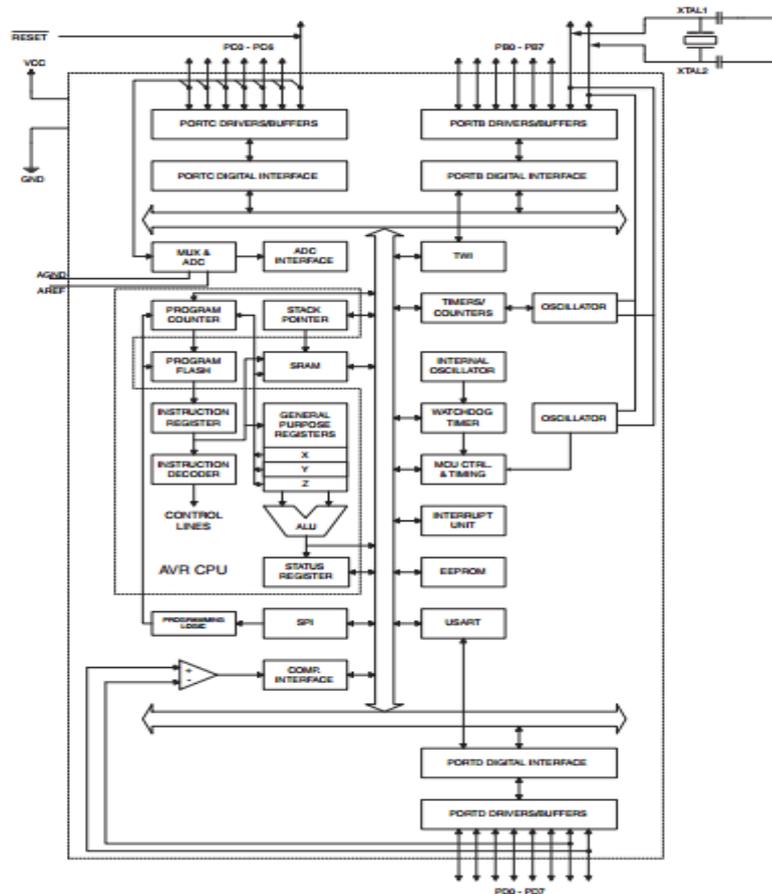


Figura 2. 5: Diagrama de bloques del Atmega 8L
Fuente: (Atmel Corporation 2013).

El Atmega8 presenta las siguientes características: 16k bytes de flash programable en el sistema con capacidad de lectura y escritura, 512 bytes de EEPROM, 1k byte de SRAM, 32 líneas de entrada y salida generales, 3 temporizadores/contadores flexibles con modos comparativos, interruptores internos y externos, un serial programable USART, un byte orientado a dos interfaces seriales de hilo, 6 canales ADC (8 canales en paquetes TQFP y QFN/MLF) con 10 bits de exactitud, un temporizador Watchdog programable

con oscilador interno, un puerto serial SPI y cinco modos de ahorro de energía seleccionables por software.

El modo ausente para el CPU permite al SRAM, temporizadores/contadores, puerto SPI e interruptores seguir laborando. El modo apagado salva los registros contenidos, congela el oscilador, deshabilita toda otra función del chip hasta la próxima interrupción o reseteo del hardware. El modo Stand by, el oscilador de cristal/resonador está corriendo mientras el resto del dispositivo está dormido. Esto permite mayor velocidad de encendido combinado con un consumo mínimo.

El ATmega8 trae incluido un conjunto completo de herramientas de desarrollo de programas y sistemas, incluyendo compiladores C, ensambladores de macros, simuladores de programas y kits de evaluación.

2.4.3. Atmega AT32UC3A

Es un microcontrolador con un sistema basado en AVR32 RISC con un procesador de núcleo de 32 bits con una frecuencia de 66 MHz, diseñado para aplicaciones embebidas sensibles al costo que tiene énfasis en el bajo consumo de energía, mayor densidad de código y mayor rendimiento. El procesador implementa una unidad de protección de memoria y un rápido y flexible controlador de interrupciones de operaciones en tiempo real y moderno que pueden alcanzar capacidades más altas usando un conjunto de instrucciones.

El controlador de acceso directo a memoria periférica (PDCA) permite la transferencia de datos entre periféricos y memoria sin implicar el procesador además reduce drásticamente la sobrecarga de procesamiento cuando se transmiten datos de streaming entre módulos.

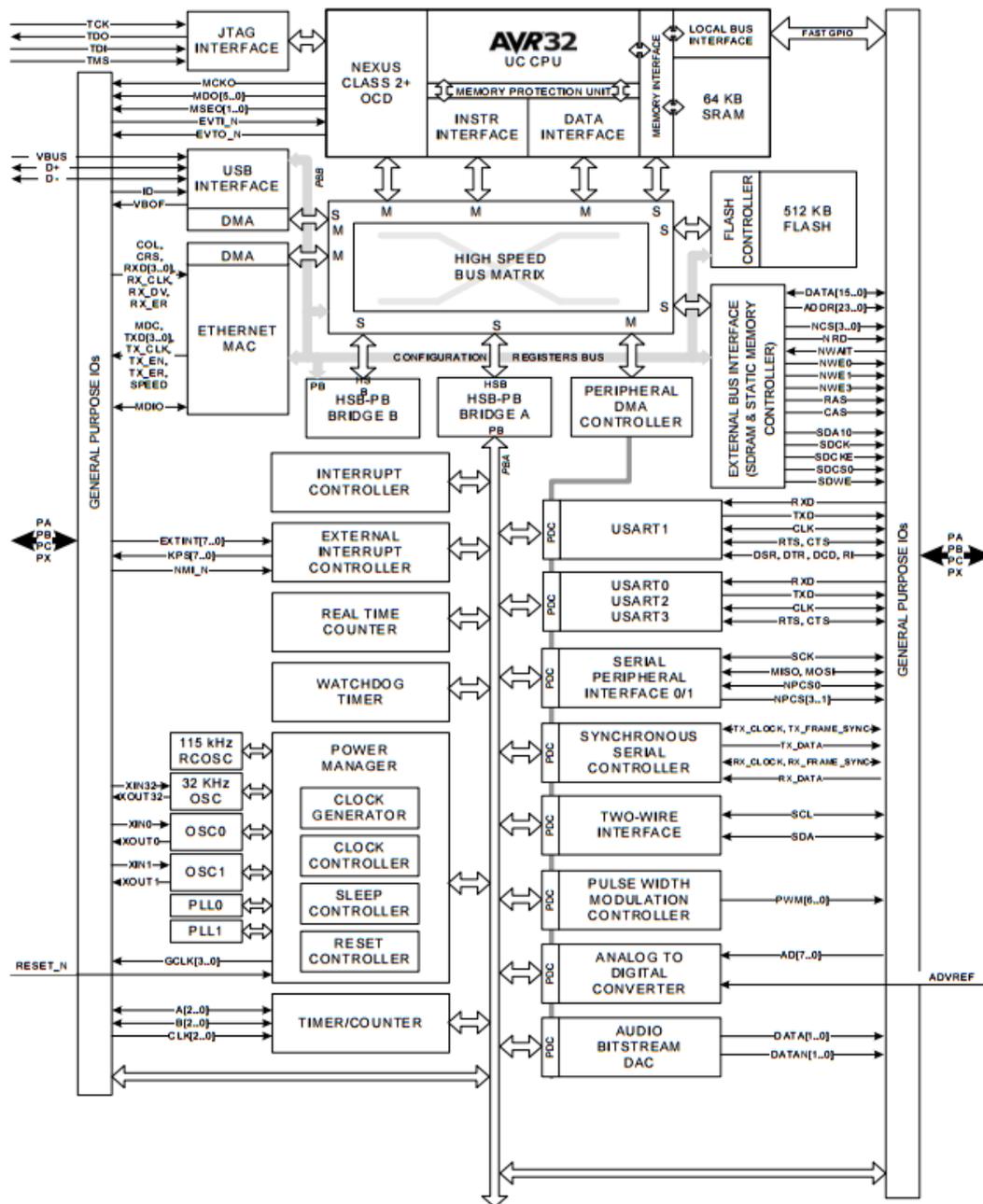


Figura 2. 6: Diagrama de bloques del Atmega AT32UC3A
Fuente: (Atmel Corporation 2012).

El AT32UC3A cuenta con 15 registros de propósitos generales y puntero de pila de 32 bits, contador de programa y un registro de vinculo que reside en la carpeta de registros, conjunto de instrucciones ortogonales, modos privilegiados y no privilegiados disponibles para la eficiencia y seguridad de los sistemas operativos, extensión DSP con aritmética de saturación y una gran variedad de instrucciones de multiplicación.

Permite una instrucción por ciclo de reloj para la mayoría de las instrucciones, 1 byte de acceso de memoria de media palabra, palabra y doble palabra, múltiples niveles de prioridad de interrupciones con MPU que permite al sistema operativo tener memoria de protección. Tiene un sistema de depuración y pruebas que tienen acceso directo a la memoria y capacidad programable mediante su interfaz, depuración extensa en el chip conforme con IEEE-ISTO 5001-2003, soporte de hardware para seis programas y dos puntos de interrupción de datos con ilimitado número de software para el soporte de interrupciones y vigilancia de puntos de seguimiento.

El sistema de bus tiene una matriz de velocidad con seis maestros y seis esclavos que anejan las solicitudes de la captura de datos del CPU, instrucción de la CPU, PDCA, controlador Ethernet, flash interno, SRAM interno, bus periférico A, bus periférico B, además soporta tres modos: no predeterminado, maestro predeterminado de acceso pasado y maestro predeterminado fijo. Ruptura con límite de ciclo de ranura y un decodificador de direcciones proporcionado por maestro.

2.5. Sistema embebido Arduino

Según K. Jácome (2016), Arduino es una plataforma de creación de prototipos de código abierto que tiene la capacidad de leer entradas y los convierten en salidas ya que es fácil de usar en hardware y software, se ha utilizado en varios proyectos que utilizan implementos científicos complejos y cotidianos y esto va a seguir evolucionando a través del tiempo.

Según Arduino (s. f.), Arduino es una herramienta fácil de usar ya que fue diseñada para estudiantes principiantes, desde su comercialización se la adaptó para que los usuarios llenen sus necesidades que varían desde simples tablas de 8 bits hasta llegar al ámbito científico.

Arduino es una plataforma de prototipos electrónicos de código abierto basados en hardware y software flexibles, a través de los años Arduino ha sido cerebro de miles de proyectos, sus contribuciones han añadido gran cantidad de conocimiento accesible que puede ser de gran ayuda para novatos y expertos.

Nació en el instituto de diseño e interacción IVREA como una herramienta fácil para prototipos sin fondos de programación y electrónica, como ha llegado a gran comunidad como placas de 8 bits hasta aplicaciones LOT, impresión 3d y ambientes embebidos, como son de código abierto dan poder al usuario independientemente y eventualmente adaptando sus necesidades particulares.

2.5.1. Arduino UNO

El Arduino UNO es una placa microcontroladora basada en el Atmega 328, consta con 14 pines digitales entrada/salida, 6 entradas analógicas, 16 MHz de un resonador de cerámica, conexión USB, conector de electricidad y un botón de reseteo. Este se diferencia de los demás ya que no cuenta con el driver USB de chip serial FTDI, pero cuenta con el Atmega 16U2 programado como un convertidor de USB a serie.

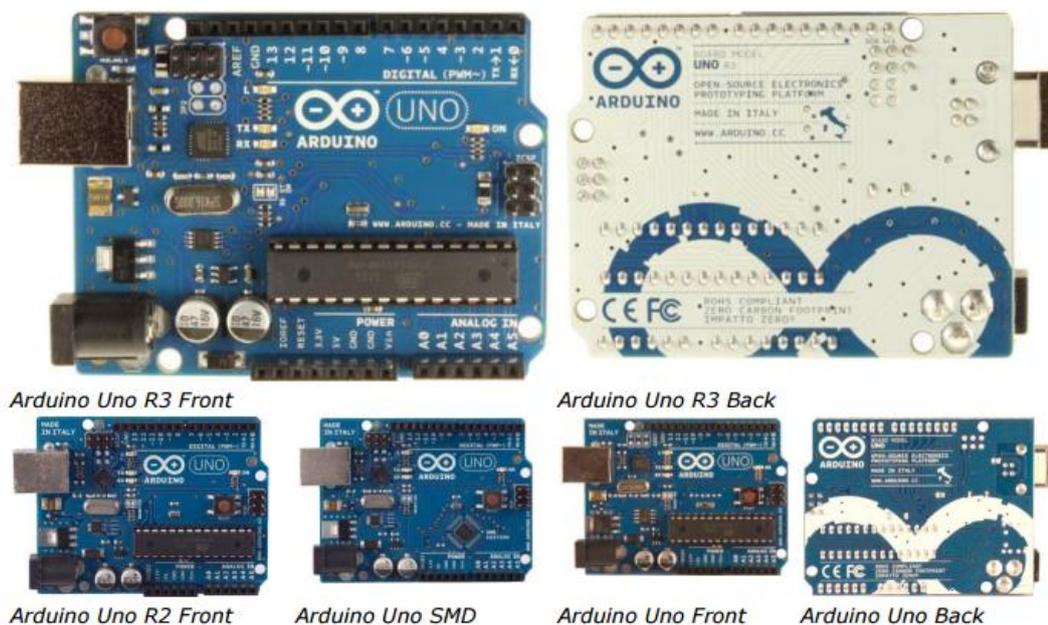


Figura 2. 7: Diferentes placas Arduino UNO

Fuente: (Farnell, 2013)

Cualquiera de los 14 pines digitales de UNO se puede dar uso dependiendo de las funciones que se les asigne, la placa funciona con 5v, los pines proveen o recibe 40 mA y cuenta con un resistor interno de 20-50 KOhms. Se puede recibir y transmitir datos serie TTL, UNO tiene 6 entradas analógicas que cada uno provee 10 bits de resolución.

Además, cuenta con un número de facilidades de comunicación con una computadora, otro Arduino o con otros microcontroladores, el software de Arduino incluye un monitor serie que permite datos de texto simples que está enviando a la placa, la librería del programa serie que tiene permite la comunicación en serie de cualquier pin digital Micro.

2.5.2. Arduino Micro

El Arduino micro es un aplaca de microcontroladores basado en Atmega 32, tiene 20 pines digitales de entrada y salida, un oscilador de cristal de 16 Mhz, una conexión de micro USB y un botón de reseteo. Contiene todo lo que se necesita para soportar el microcontrolador.

El micro está construido sobre comunicación USB, eliminando la necesidad de un procesador secundario, esto permite a micro aparecer en un computador al momento que se conecta como un mouse o un teclado.



Figura 2. 8: Placa del Arduino Micro frontal/posterior.

Fuente: (Farnell, 2013)

El micro consta con 20 pines digitales que se pueden utilizar como entrada y salida, cualquier pin provee 40 mA y constan con un resistor interno de 20-50 KOhms, además permite una comunicación serial sobre USB y

aparece como un puerto virtual en el software de la computadora y actúa como un dispositivo USB 2.0 de rápida velocidad, usando drivers estándares USB COM.

2.5.3. Arduino Nano

El Arduino nano es una pequeña pero amigable base para tablero de circuitos microcontroladores sobre Atmega 328 o Atmega 168, tiene similitud con el Arduino Dueminalove, pero en un paquete distinto, trabaja con un mini cable y con corriente continua elaborado por Gravitech.

Este embebido tiene un número de facilidades de comunicación con la computadora, otro Arduino o con otros microcontroladores, poseen pines digitales '0' para Rx y '1' para Tx. Provee un puerto virtual del software en la máquina. El software de Arduino nano incluye un monitor serie que permite datos de textos simples que está enviando al tablero Arduino, los leds RX y TX en el tablero que parpadean cuando se transmiten los datos por medio del chip FDTP y la conexión USB de la máquina.

Cualquiera de los 14 pines digitales del Arduino nano se puede usar para entrada y salida, cualquier puerto provee o recibe un máximo de 40mA y tiene un resistor interno de 20-50 KOhms. El Arduino nano tiene facilidades de comunicación con computadoras u otros microcontroladores ya que poseen una serie de comunicaciones UART, la placa (ver figura 2.9) canaliza esta comunicación serial a través de USB y controladores FTDI.

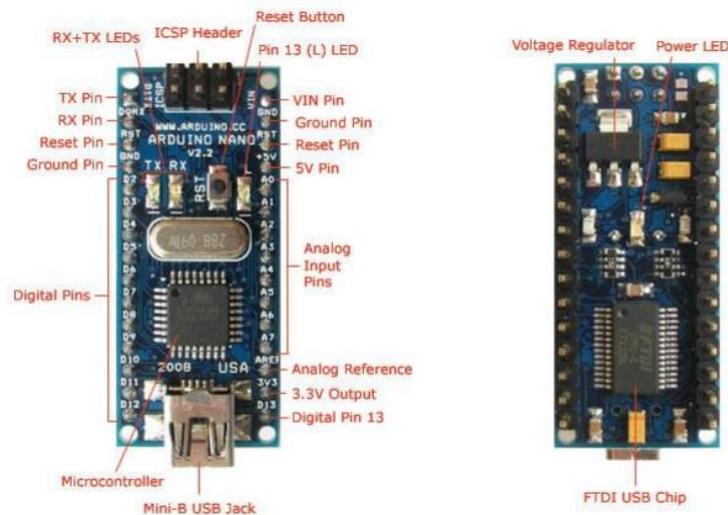


Figura 2. 9: Esquema de tarjeta Arduino Nano
Fuente: (Farnell, 2013)

2.6. Comunicaciones por radiofrecuencia.

Según M. Stoehr (2012), las señales de radiofrecuencia son una forma de onda electromagnética como la luz visible, que constituyen una parte del espectro, este abarca todas las formas de luz que van desde frecuencias de audio como la omnipresente de 60 HZ, a través de las bandas de radio que incluyen AM, FM, canales de televisión y otras bandas de radiofrecuencia.

Según G. Arimany (2011), las radiofrecuencias es el nuevo cambio en robots modulares. La ventaja central de esta tecnología comparada con otros es que nos permite la comunicación desde un rango de corto y mediano alcance, puede cruzar obstáculos, no se pierde la comunicación si hay un desalineamiento y no necesita un módulo o hosts para comunicarse.

El componente central de las radiofrecuencias es el transmisor, el cual modifica la señal original la cual es adecuada por el transmisor, el significado de transmisión se basa en que la señal viaja entre el transmisor y el receptor, los cuales transforman la información dentro de la señal original que se procesa después.

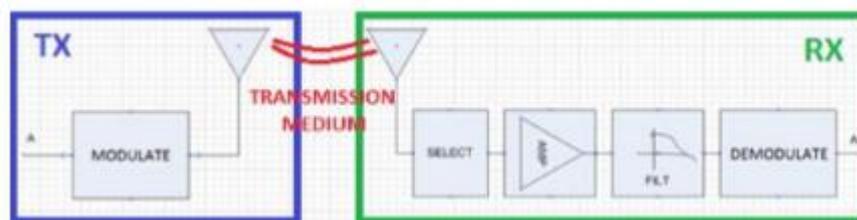


Figura 2. 10: Principales componentes para implementar un sistema de comunicación RF.

Fuente: (G. Arimany, 2011).

Para poder implementar un sistema de comunicación de radiofrecuencia es esencial implementar cada componente, sin embargo, hay equipos que tienen los circuitos del transmisor y receptor combinados, llamados transceiver, en el cual se implementan las características explicadas anteriormente. De todas formas, para diseñar un buen sistema de RF, es necesario tomar en cuenta varias variables de la frecuencia en que trabaja o las técnicas que usa para modular/ desmodular la señal.

Hay diferentes tipos de clasificación de comunicaciones inalámbricas RF. Un camino es si el dispositivo cumple con los protocolos estándares o no, otra clasificación es acorde con la frecuencia de operación en que trabaja el dispositivo.

Algunos dispositivos no cumplen con los protocolos estándares, pero la mayoría de ellos lo hacen ya que existen diferentes protocolos que funcionan en diferentes rangos de frecuencia, la frecuencia de operación son algunas bandas de frecuencia las cuales no necesitan licencia si estas no superan los límites que se encuentran menor que 1Ghz estos son usados desde 300Mhz hasta 900 MHz y 2.4 GHz son normalizadas en el mundo.

La técnica utilizada en radiofrecuencia por lo general es la banda estrecha o banda ancha que utiliza todo el ancho de banda disponible en lugar de utilizar una señal portadora por lo que esta técnica es útil para tener una comunicación confiable.

En los robots los módulos son muy cerrados uno del otro, por esta razón las tecnologías RSSI es una medida de la potencia presente en el receptor que normalmente se mide en el amplificador y LQI que es una medida de la calidad de la señal recibida, cuanto mayor sea el LQI, el rango es mejor, este parámetro suele estar vinculado a RSSI porque una señal potente será menos afectada por el ruido; son muy utilizadas para detectar si la señal recibida es de un módulo vecino o de algún otro módulo.

CAPÍTULO 3: Diseño e Implementación.

En este segmento se detallará la implementación física de todo el sistema que conforma nuestro robot Minisumo RC y Autónomo. Para ello es necesario la aplicación de varios conceptos básicos conocidos en el transcurso de la formación académica.

3.1. Principio de funcionamiento

El robot minisumo RC y autónomo fue considerado para su aplicación en los nuevos estilos de competencia realizada inicialmente en Colombia, el cual se tuvo una participación con resultados satisfactorios. El Robot está diseñado para participación de ambas categorías, tanto minisumo RC y autónomo en concursos nacionales e internacionales.

La categoría de minisumo RC se lleva implementando en varios países desde hace un tiempo atrás, pero en Ecuador no se ha llegado a su aplicación. Por el contrario, la categoría autónoma ya se lleva aplicando por más de diez años en el país y muchos más de manera internacional. La competencia de mini sumo RC es parecida a mega sumo RC solo varían en tamaño, el material del dojo y su peso siendo este un máximo de 500 gramos, en esta categoría será necesaria la operación por mando a distancia conectada por un enlace de radiofrecuencia al robot para realizar movimientos de esquite y ataque contra el adversario.

En la categoría de minisumo autónomo se realiza de manera independiente, no es legal la operación por medio de un mando a distancia ya que realiza la lectura digital o analógica de los sensores que se encuentren montados en la estructura, los movimientos serán efectuados gracias a la evaluación de dichos resultados y por medio de un algoritmo se especifica la dirección, sentido y fuerza.

Entre el hardware utilizado para nuestra aplicación se encuentran sensores, motores, microcontroladores, módulos bluetooth y otros que se detallan a continuación.

3.2. Hardware

Para la implementación del robot que sea autónomo y controlado por radiofrecuencia es necesario analizar varios dispositivos y determinamos cuales fueron los necesarios.

3.2.1. Sensor OMRON MR45 difuso

Este dispositivo es utilizado ampliamente para la detección de objetos a una determinada distancia que por cierto es variable y depende de una previa configuración. El sensor, figura 3.1, es capaz de detectar un objeto gracias a la reflexión del emisor infrarrojo que encuentra en él, y gracias a un receptor se determina un valor digital en alto (5v) o en bajo (0v) según la respuesta, gracias a ello se puede proceder a su aplicación en microcontroladores.

Este posee un resistor variable para determinar la distancia a la cual se desea trabajar, con la configuración apropiada se puede detectar obstáculos desde 1 cm a 45 cm.



Figura 3. 1: Sensor Omron MR 45
Fuente: (Dede, 2016)

3.2.2. Sensores de reflexión IR

Estos sensores son muy utilizados en varias categorías de concursos de robótica a nivel nacional e internacional, su funcionalidad es simple, se encarga de reconocer un color determinado y poder así enviar un valor digital a su salida para ser receptada por un sistema de control que permita determinar decisiones con algún algoritmo implementado.

Para ser específicos en su aplicación en la categoría de minisumo autónomo, estos ayudan a la detección del borde de color blanco que rodea el dohyo de competición, con ello el robot gracias a su algoritmo implementado en el microcontrolador evitará salir de dicho escenario al momento de atacar o ser atacado por un contrincante.

Es conocido que el color negro absorbe mayor cantidad de luz y el blanco la refleja. Se emite un haz de luz infrarrojo y de este fenómeno descrito se genera un valor digital de salida. Existen muchos tipos de sensores que tienen

la misma funcionalidad, pero por experiencia en otras competencias se decidió utilizar los sensores QRT-1RC de la empresa Pololu como se muestra en la figura 3.2.



Figura 3. 2: Sensores Pololu QRT-1RC
Fuente: (Pololu, 2016)

Este sensor Pololu QRT-1RC, posee un diodo emisor de infrarrojos y un fototransistor para detectar el reflejo, es así cuando el porcentaje de reflexión es de menor o mayor manera se interpretará como un color más oscuro o más claro. Se realizaron pruebas y se comprobó que este dispositivo debe colocarse a una distancia no mayor de los 0.5mm sobre la superficie para que se tenga los mejores resultados en la competencia. En la figura 3.3 se muestra el esquemático electrónico proporcionado por el fabricante Pololu.

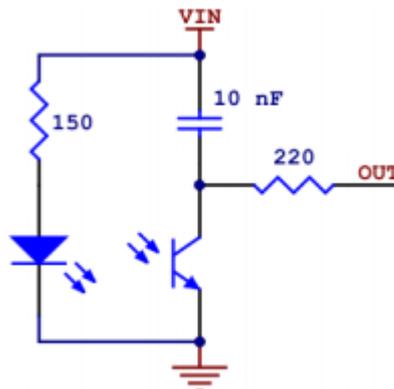


Figura 3. 3: Esquemático del sensor Pololu QTR-1RC
Fuente: (Pololu, 2016)

3.2.3. Motores fingertech robotics spark gearmotor

En nuestro prototipo es obligatorio emplear motores de mayor torque y que puedan responder a las exigencias que solicita en cada competencia para ello fue necesario recurrir a los conocidos motores del fabricante FINGERTECH ROBOTICS, el cual ofrece una gran variedad de dispositivos para los robots que se desea construir.

Entre las opciones proporcionadas por esta reconocida marca se seleccionó los motores FINGERTECH ROBOTICS PARK GEARMOTOR con una relación de 33.3:1 por su velocidad y torque. Se muestra en la figura 3.4, el dispositivo que se llegó a emplear en nuestro prototipo.



Figura 3. 4: Motor DC Spark con caja reductora.
Fuente: (FingerTech Robotics, 2016)

Adicional, podemos incluir la curva característica proporcionada por su fabricante en la siguiente figura 3.5 el cual nos sirvió de guía para analizar cuan eficiente deberá responder nuestro prototipo en la competencia.

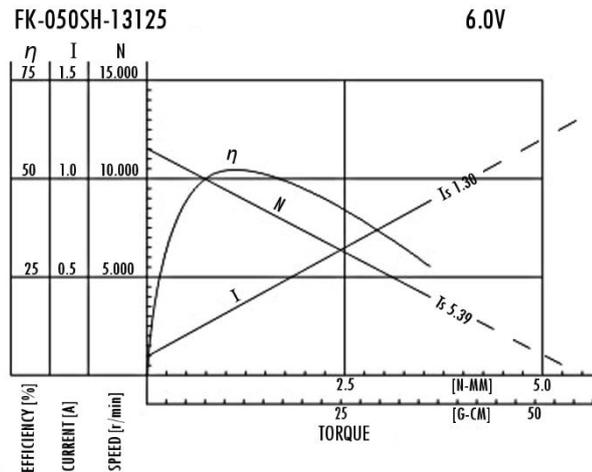


Figura 3. 5: Curva característica de los motores Spark Gearmotor
Fuente: (FingerTech Robotics, 2016)

3.2.4. Placa Arduino Pro Micro

En el sistema de control de nuestro robot es necesario emplear alguna forma de control por medio de microcontroladores, para ello fue necesario seleccionar de entre una gran variedad de sistemas embebidos uno que se adaptara a nuestra necesidad, entre ellas fueron velocidad de procesamiento, tamaño, facilidad de uso y programación, entre otros.

El sistema utilizado fue el de la ya conocida placa de sistema embebido Arduino por su gran variedad de librerías existentes y su facilidad de aplicación tanto en hardware y software. La placa de Arduino seleccionada por su tamaño fue el modelo Pro Micro, este posee un microcontrolador ATmega32U4 y los pines necesarios para nuestro proyecto los cuales se pueden configurar como entradas y salidas digitales y otros como analógicas.

En la siguiente figura 3.6 se presenta el microcontrolador ATmega32U4, que viene incorporado en la placa embebida Arduino.

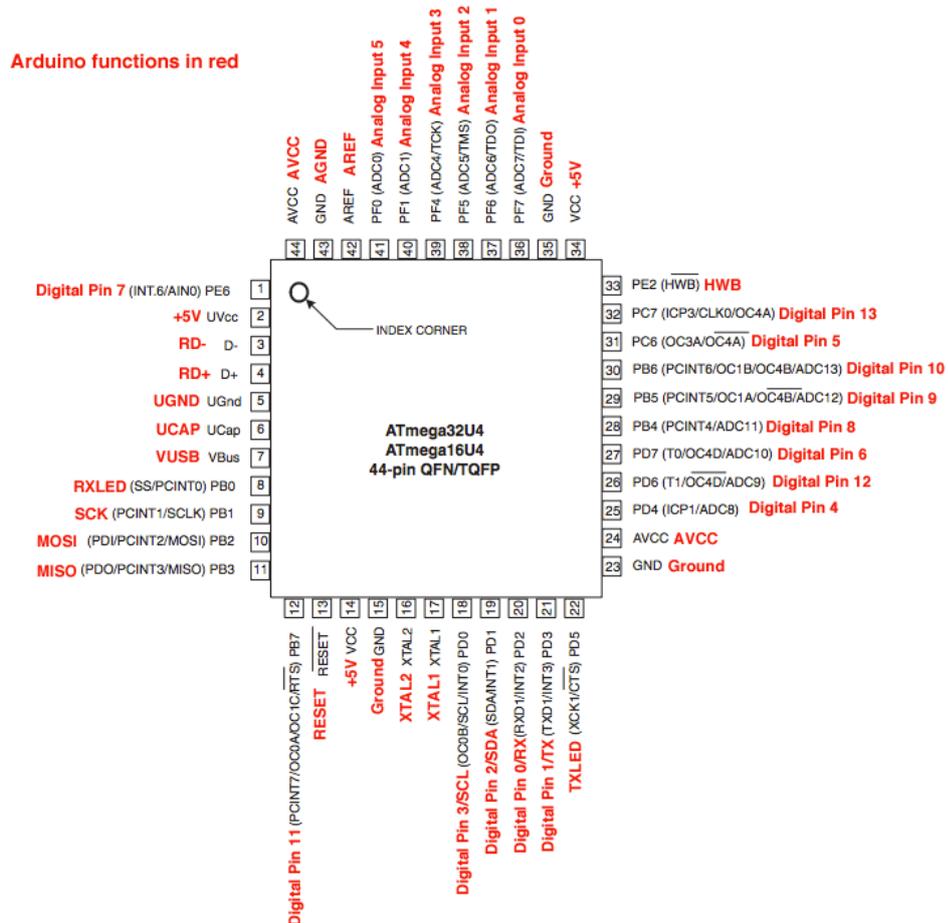


Figura 3. 6: Pines de microcontrolador ATmega32u4
Fuente: (Arduino, 2016)

En la wiki del módulo Arduino Pro Micro podemos encontrar en detalle la imagen 3.7 correspondiente al pinout, con ella es fácil referenciar como se llegará a construir la placa madre de nuestro robot en donde se incorporará y formará parte de un sistema completo entre la comunicación por radio ofrecida por otro modulo y el control de los motores con un driver de alto rendimiento.

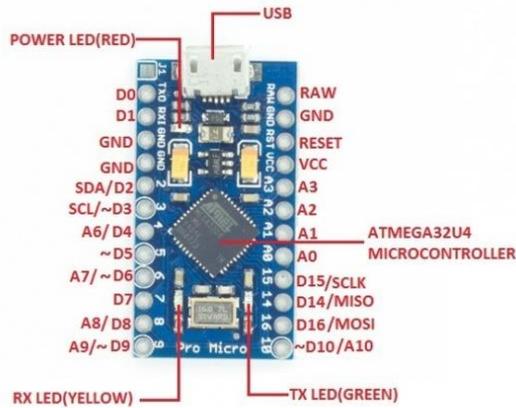


Figura 3. 7: Pines de Arduino Pro Micro
Fuente: (Arduino, 2016)

Para la programación de su algoritmo se utilizará el IDE de Arduino, figura 3.8, una interfaz creada para el uso de este tipo de sistemas embebidos y con ella se logrará la carga del software a la placa.

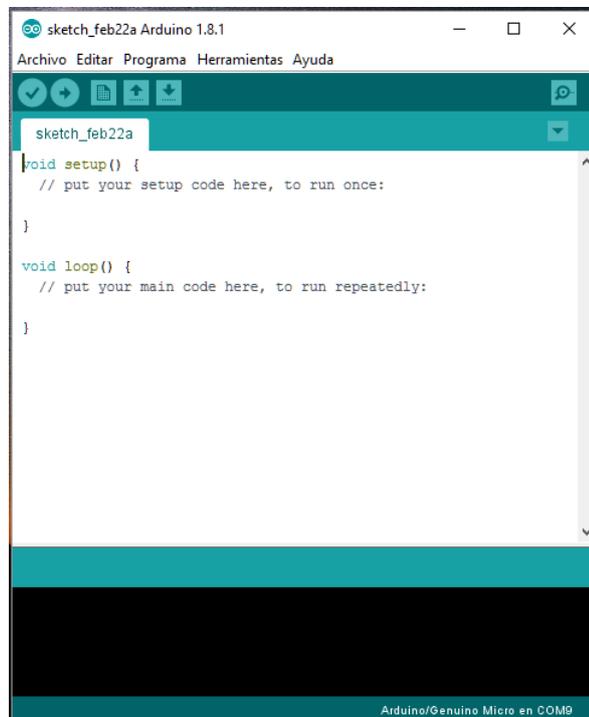


Figura 3. 8: IDE Arduino
Elaborado por: Autores.

3.2.5. Módulo de comunicación Bluetooth HC-05

Para las comunicaciones de radiofrecuencia que permitirá a la placa arduino recibir los comandos que permitirán controlar el robot se utilizará un módulo de comunicaciones bluetooth HC-05 ya que posee las características necesarias para su incorporación al proyecto y su facilidad de adquisición dentro del mercado nacional.

El módulo permitirá por medio de radiofrecuencia comunicar de forma inalámbrica un Smartphone y la placa arduino por medio de una aplicación que se desarrollará desde cero en este proyecto. En la siguiente figura 3.9 se muestra el módulo que se montara en la placa madre del controlador del robot minisumo.

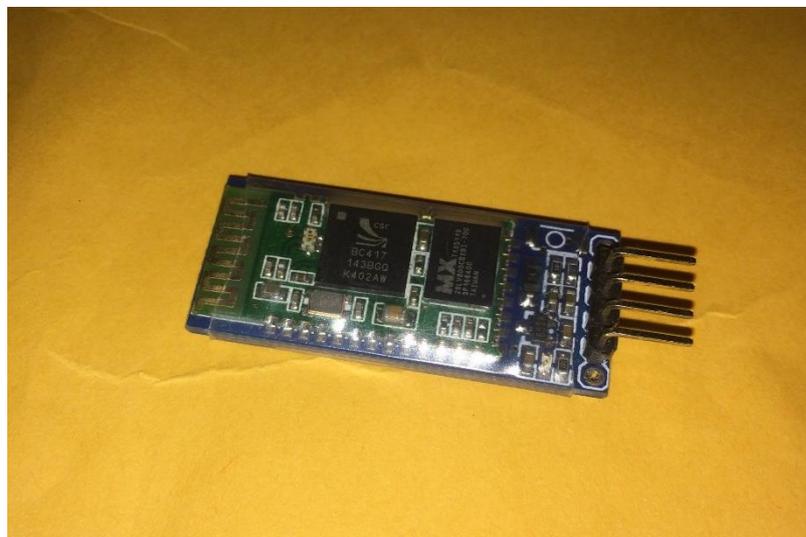


Figura 3. 9: Módulo bluetooth HC-05
Elaborado por: Autores

Como se pudo ver en la figura anterior, este módulo posee cuatro pines, dos de comunicación serial que se recibirían los datos transmitidos por el

Smartphone o Tablet con sistema operativo Android, y dos pines que alimentarán con 5V y GND. En la tabla 3.1 se detallan las características principales de este dispositivo.

Tabla 3. 1: Características principales del módulo Bluetooth HC-05

Características	Detalles
Tensión de alimentación	3.3 – 6 voltios
Consumo de corriente máximo	40mA
Consumo de corriente mínimo	8mA
Modulación	GFSK (Modulación por Desplazamiento de Frecuencia Gaussiana)
Frecuencia	2.5 GHz en las bandas ISM
Potencia de salida	2.5 mW – 4 dBm
Velocidad de transmisión de datos	3 Mbps
Dimensiones	26.9mm x 13mm x 2.2mm

Elaborado por: Autores.

3.2.6. Controlador de motores de alto rendimiento Sabertooth 2x5

Para controlar el giro de los motores descritos en este capítulo anteriormente es necesario adquirir un controlador de motores de alto rendimiento y a la vez pequeño para que pueda encajar con la placa madre que se construirá. Definiendo las características necesarias se selecciona la tarjeta Sabertooth 2x5 A por su costo y beneficio que proporciona dicha tarjeta, en la figura 3.10 se muestra este dispositivo.

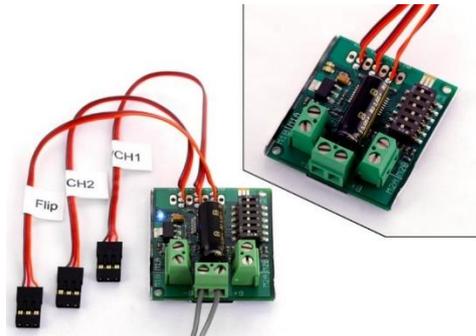


Figura 3. 10: Sabertooth 2x5
Elaborado por: Autores

Según el fabricante este dispositivo puede alimentar dos motores y proporcionar a cada canal con 5 amperios y 10 amperios pico, tienen protección contra la sobretensión y temperatura el cual ayudará a mantener la vida útil de la placa, su alimentación puede ser desde 6v hasta los 18v, su peso es de apenas de 18 gramos y sus dimensiones son de 45 x 40 x 13 mm.

Con las características descritas anteriormente hacen que este dispositivo por su tamaño y gran eficiencia sea el referente necesario para poder aplicarlo en nuestro prototipo ofreciendo así a los motores de alto torque la potencia necesaria para poder desempeñar su máximo rendimiento.

3.2.7. Baterías LiPo

En competencias nacionales o internacionales los prototipos exigen mucho de sí, y para ello es necesario poseer una alimentación de energía que permita al robot dar todo su potencial. Entre los tipos de alimentación que existen en el mercado se eligió el uso de baterías de polímero de litio (LiPo)

ya que, por sus características eléctricas, su tamaño y peso fueron el excelente candidato para ser usadas dentro de un robot.

Entre sus características eléctricas están su tasa de descargar “C”, su amperaje y las celdas que poseen para proporcionar un determinado voltaje. La batería que se utilizará en nuestro prototipo es de 3 celdas en serie proporcionando un voltaje de 12V DC, el amperaje es de 1500 mAh y una tasa de descarga de 40 C como se puede apreciar en la figura 3.11 a continuación.



Figura 3. 11: Batería LiPo
Elaborado por: Autores.

3.2.8. Diseño electrónico

A continuación, se presenta los detalles del diseño electrónico de nuestro prototipo robot minisumo controlado por mando a distancia y autónomo. Para realizar el esquemático se utilizó el programa Proteus 8, en el cual se agregaron las librerías correspondientes a los módulos Arduinos y se simuló con el dispositivo más parecido al que se aplicará en la práctica. El diseño consta de tres etapas, alimentación, obtención de datos por sensores o por el puerto serial y control de movimientos gracias a la Sabertooth.

En la figura 3.12 que se muestra a continuación, esta detallado la etapa de alimentación, este proviene de una batería de lipo a la que será conectado a una bornera y pasara por un regulador 7805 que nos facilitará la regulación del voltaje a los 5 voltios necesarios para que se accionen los sensores, el módulo bluetooth y por supuesto la placa arduino.

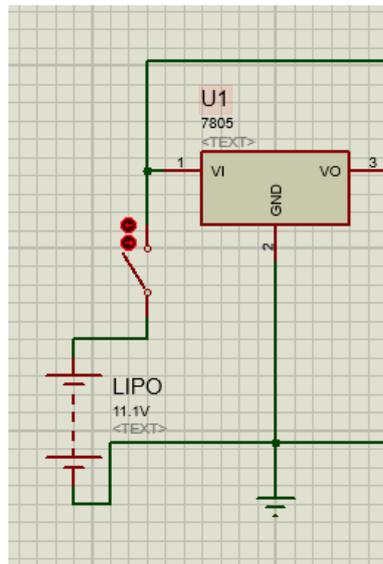


Figura 3. 12: Etapa de alimentación
Elaborado por: Autores.

Cabe destacar que por efectos de simulación se decidió utilizar una batería de la librería y asignarle el voltaje equivalente a una Lipo de 3 celdas (3S). Esta alimentación que para por la etapa de reducción a un voltaje moderado de 5V ya que los dispositivos como los sensores, la placa arduino y el modulo bluetooth no se puede exceder de este valor ya que automáticamente dejarán de trabajar. La etapa de la obtención de datos transmitidos inalámbricamente y recibidos por el puerto serial y la lectura de los sensores están representados en la parte no sombreada de la figura 3.13.

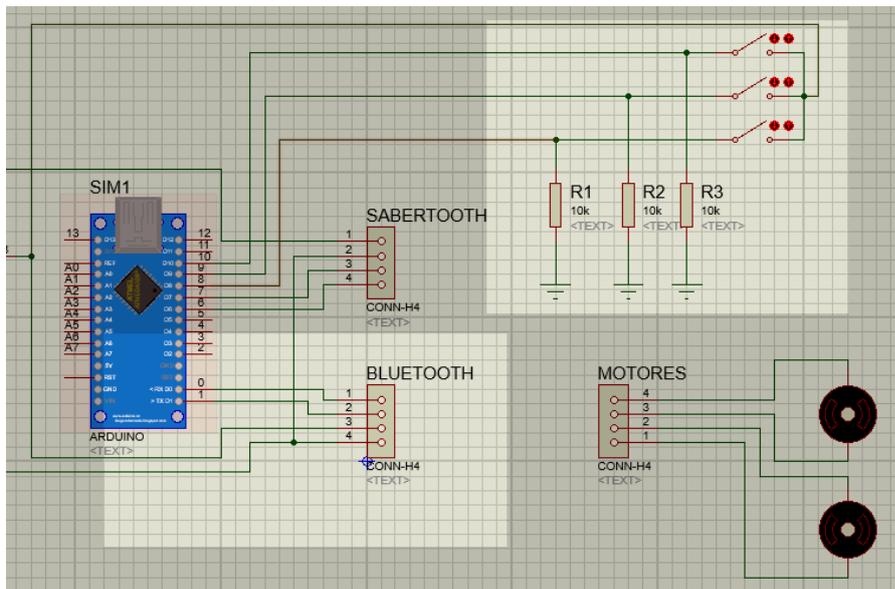


Figura 3. 13: Etapa de obtención de datos por sensores y serial
Elaborado por: Autor

Los puertos TX y RX del módulo bluetooth se conectan en dos puertos que se pueden designar en el momento de programar la placa arduino para recibir datos vía serial, y los sensores tienen que ir conectados a puertos digitales de entrada y salida de la placa. Los sensores están representados por switch ya que estos al momento de realizar sus funciones lo único que envían como dato es un valor digital alto (5V) o en bajo (0V).

En el código fuente de la programación se detallará más adelante como se tomarán en cuenta estos pulsos digitales para realizar maniobras gracias a la Sabertooth. Finalmente, el control de los motores se realizará por medio de un controlador Sabertooth 2x5, los pines de salida deberán ser pines que permitan enviar señales PWM, estos ayudarán mediante su programación a determinar la dirección de sentido de giro y su rapidez como se llega a ver en la figura 3.14.

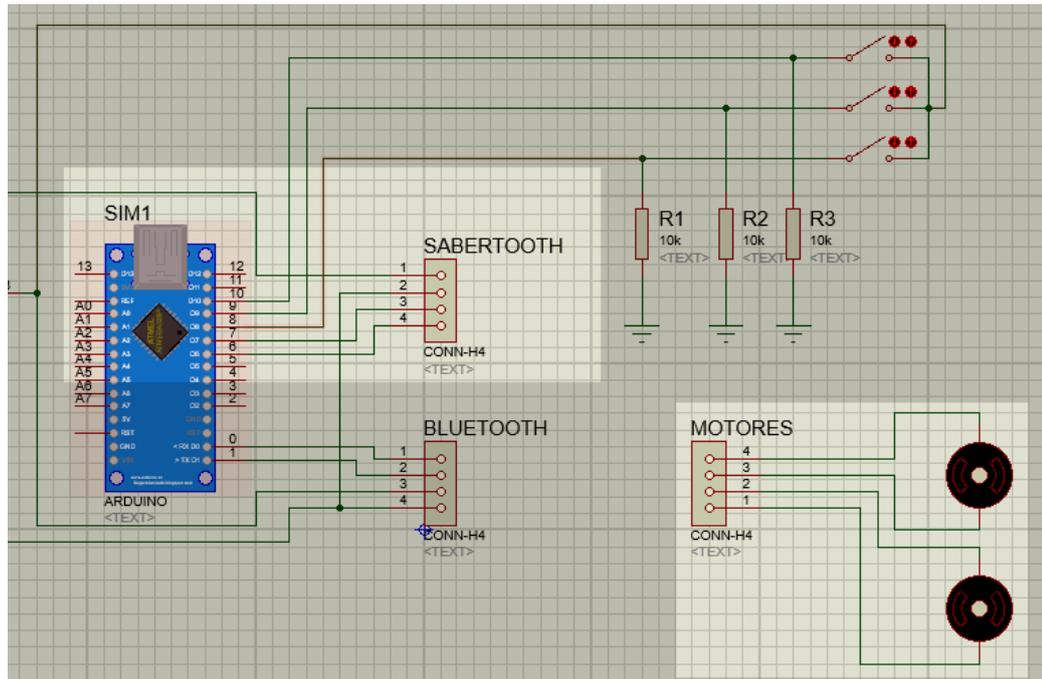


Figura 3. 14: Etapa de control de motores
Elaborado por: Autores.

A continuación, en la figura 3.15 se podrá apreciar de manera general todo el diseño esquemático para realizar su respectiva simulación y en la figura 3.16 su prueba en un protoboard.

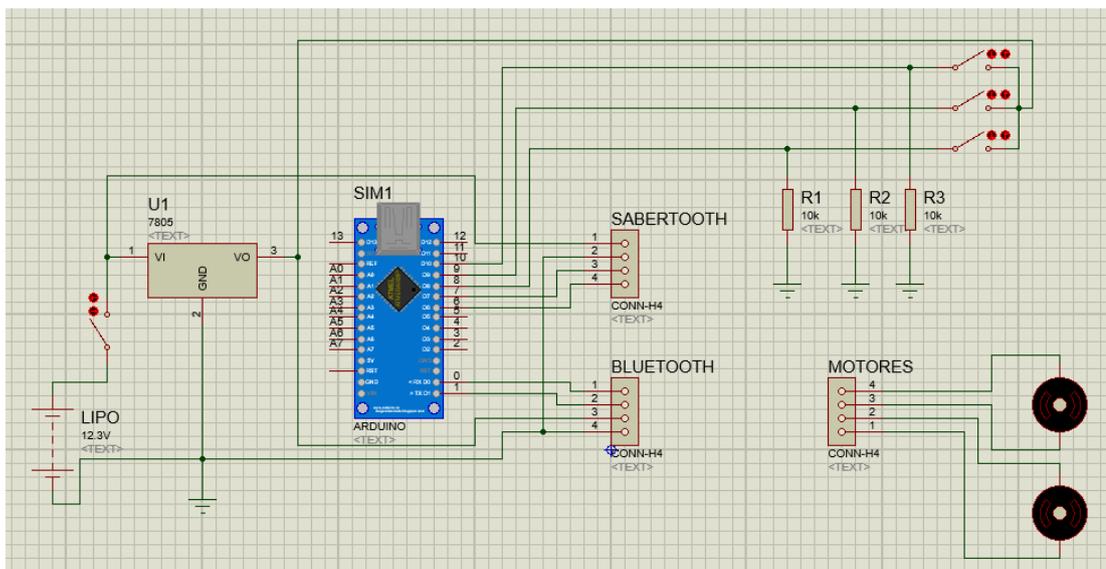


Figura 3. 15: Diseño esquemático del circuito.
Elaborado por: Autores.

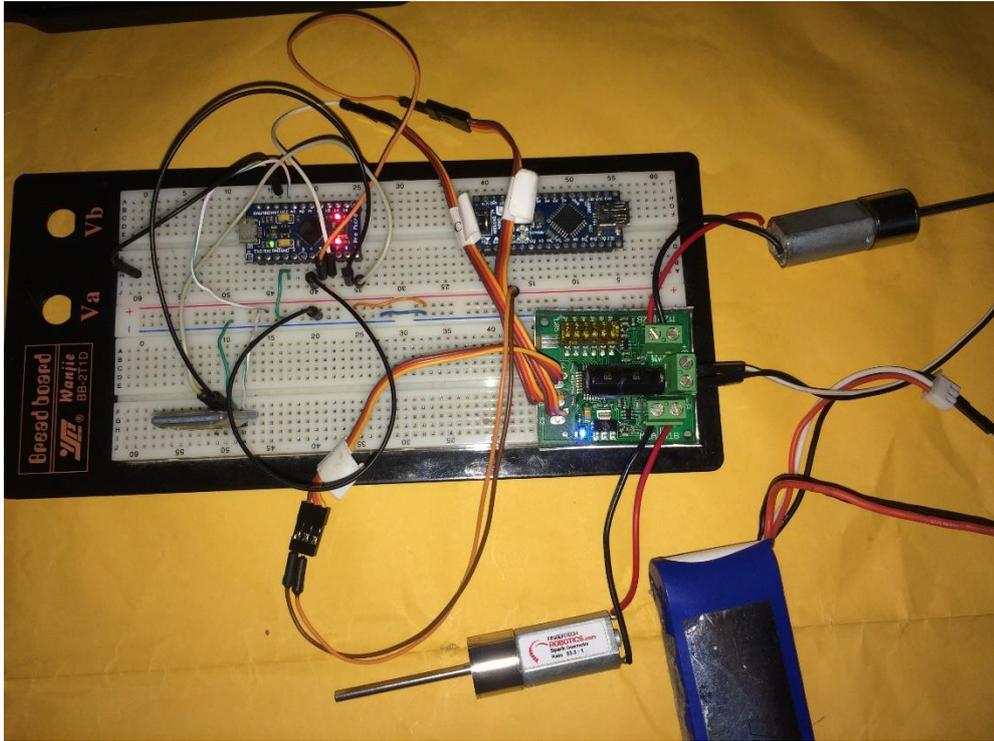


Figura 3. 16: Prueba de simulación.
Elaborado por: Autores.

Finalmente, cuando se haya realizado las pruebas del circuito en la simulación y en protoboard se procederá a realizar el diseño PCB utilizando el mismo programa Proteus en el apartado de Ares, el cual nos ayudará a definir el tamaño y características principales de nuestro circuito.

A continuación, la figura 3.17 podemos notar los detalles de cómo las pistas son utilizadas para la interconexión de los módulos que integraran este sistema.

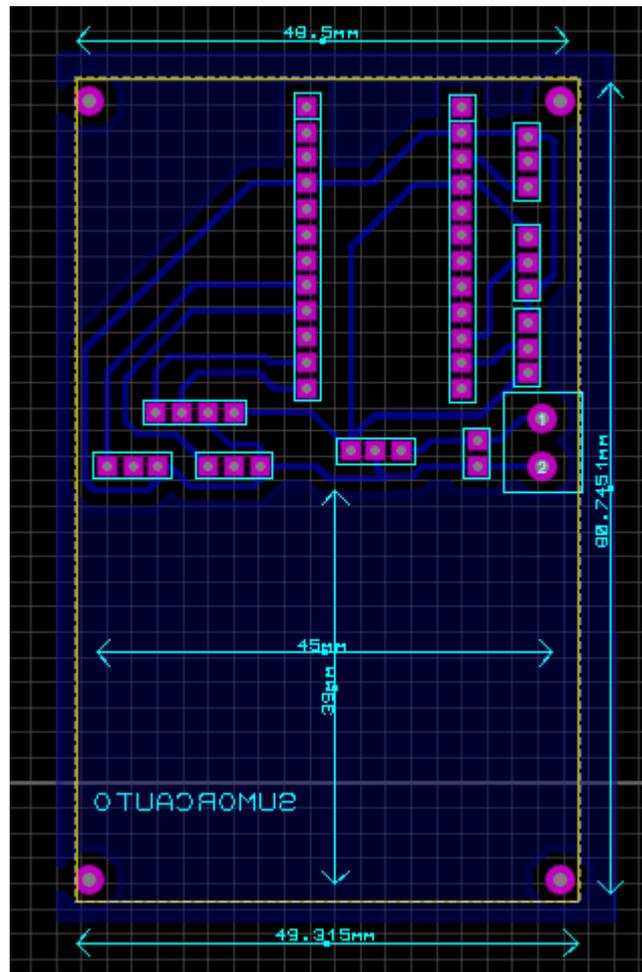


Figura 3. 17: Diseño de PCB del robot minimo
Elaborado por: Autor

3.2.9. Diseño estructural.

Se procede a realizar el diseño estructural del robot, para ello es necesario tener en cuenta el tamaño de los motores y las ruedas que utilizarán un aro de plástico para poder ser envuelta por un material antideslizante como es el caucho de silicona. El tamaño del tambor del motor es de 42mm de largo con un diámetro de 15mm, las ruedas tienen un tamaño de 21mm de largo con un diámetro de 20mm.

Con estas medidas es imposible que se pueda cumplir el tamaño reglamentario, para ello se resolvió utilizar los tambores de forma inversa a lo que se usa regularmente, cubriendo así parte del tambor por la rueda misma, el eje restante se lo envía a cortar en un taller torno. La placa inferior por el que reposaran los motores estará hecha de acrílico, y en ella también descansa la cuchilla de aluminio construida en un taller con las especificaciones requeridas.

En la figura 3.18 se podrá observar el resultado de la base, la cuchilla y los motores montados sobre una placa, esta adicional da espacio para poder montar una batería de 500mAh o más, y el soporte para empotrar la placa madre del diseño electrónico previamente descrito.



Figura 3. 18: Diseño estructural del minisumo
Elaborado por: Autores.

En la figura anterior se muestra el trabajo final del diseño estructural, este garantizará su durabilidad en competencia de gran nivel para la categoría minisumo RC o Bluetooth, así como también en la categoría minisumo autónomo.

3.3. Software Implementado

Al terminar la construcción de la placa madre donde se encontrarán interconectados los distintos módulos y su estructura física que soportará los choques y movimientos bruscos de ataque y esquite en la competencia se procederá a la creación de un algoritmo que permita que nuestro prototipo sea controlado por una aplicación en un Smartphone y de manera autónoma también funcionar.

3.3.1. Creación de aplicación SUMOAPP

Esta aplicación nos permitirá dar solución a dos eventos:

- El conteo de los cinco segundos al iniciar una competencia de manera autónoma y poder parar toda actividad del robot cuando el juez nos indique.
- El control por medio de un mando dentro de la aplicación para controlar el robot de manera inalámbrica en la competencia de minisumo RC.

Para la creación de nuestra aplicación que funcionara sobre un Smartphone o Tablet con sistema operativo Android se utilizó el aplicativo online App Inventor, en la figura 3.19 se presenta dicha herramienta.

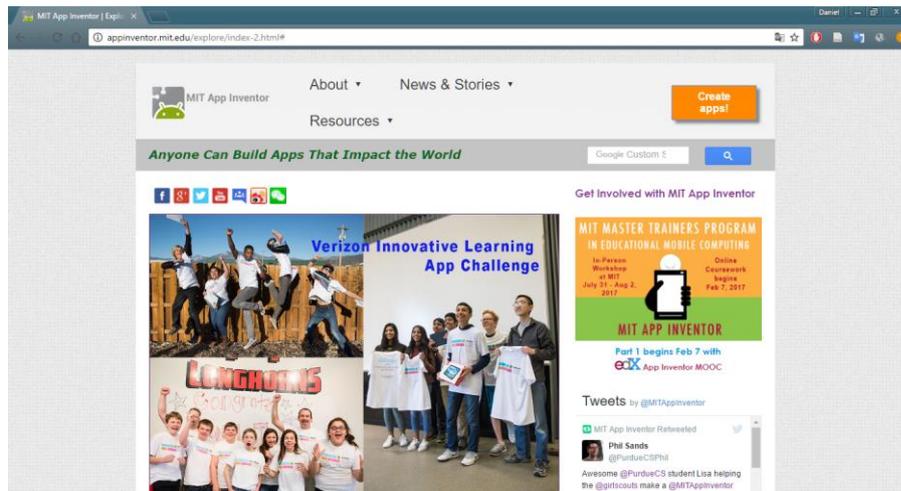


Figura 3. 19: Aplicativo online App Inventor
Elaborado por: Autores.

En la página que se muestra en la imagen anterior debemos dar click en crear app para iniciar el desarrollo de aplicaciones en Arduino. Posteriormente, se abrirá otra ventana que nos permitirá administrar todos nuestros proyectos como se puede apreciar en la figura 3.20.

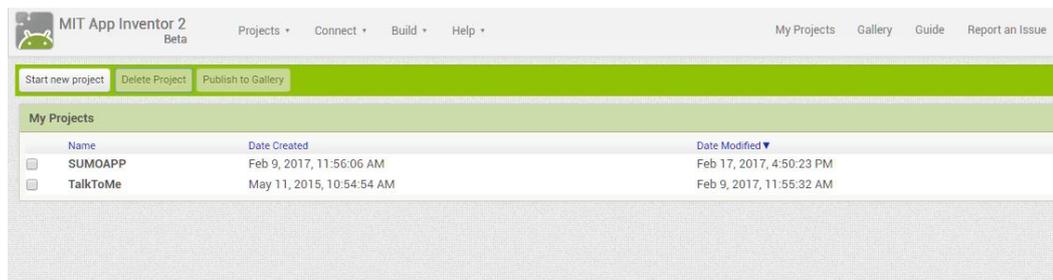


Figura 3. 20: Administración de proyectos de App Inventor
Elaborado por: Autores.

Nuestra aplicación SUMOAPP estará conformada por tres pantallas, la principal que nos permitirá elegir el modo de competición (RC o Autónomo) y según la opción seleccionada se abrirá a la que corresponda. En la figura 3.21 se aprecia la pantalla principal en el que se encuentran un conjunto de herramientas que nos permiten crear dicha app.

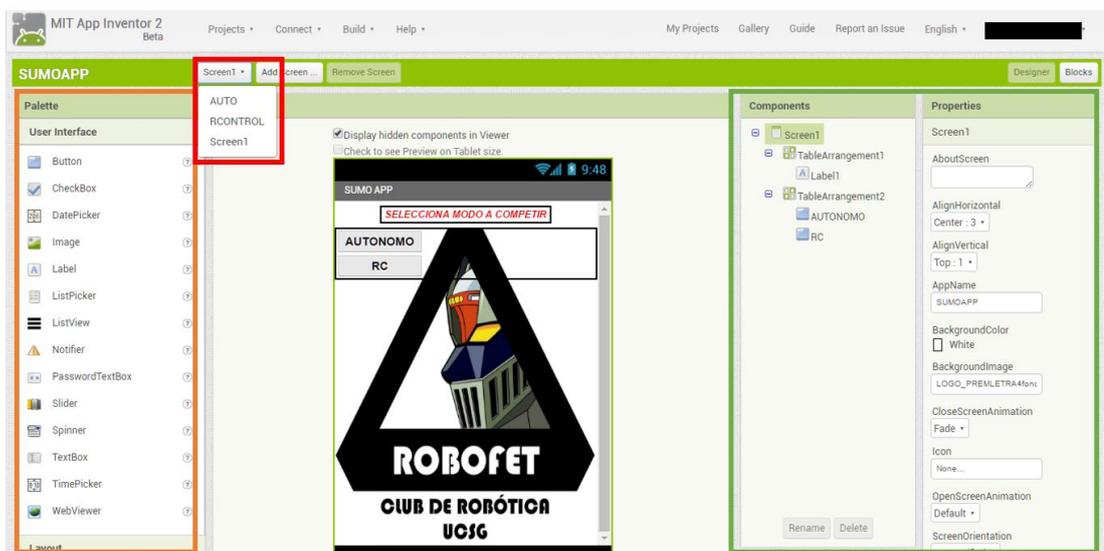


Figura 3. 21: Pantalla de herramientas para construcción de app.
Elaborado por: Autores.

En la imagen anterior se puede apreciar lo que se encuentra enmarcado con color naranja los componentes de una aplicación que pueden ser agregadas a la pantalla principal. Lo que se encuentra con rojo son las pantallas de una aplicación, como indicamos anteriormente nuestro aplicativo deberá tener tres pantallas, una de menú principal y dos para los modos de competencia.

Y por último los componentes y las propiedades se encuentran enmarcadas de color verde, en esta área es posible editar color, tamaño de letra, fondos, y muchas características de cada botón, textbox, notificación, capa, etc., que está agregado a nuestra aplicación. A continuación, se presentará en las figuras correspondientes cada una de las pantallas de nuestra aplicación.

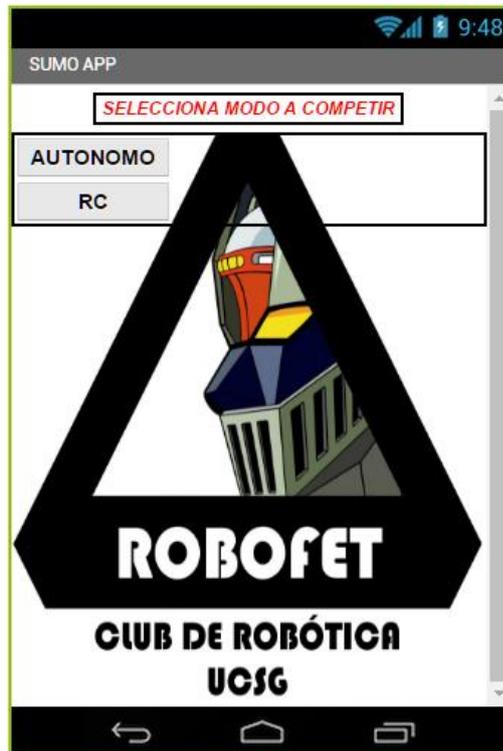


Figura 3. 22: Pantalla principal de SUMOAPP.
Elaborado por: Autor



Figura 3. 23: Pantalla de opción autónomo de SUMOAPP.
Elaborado por: Autor

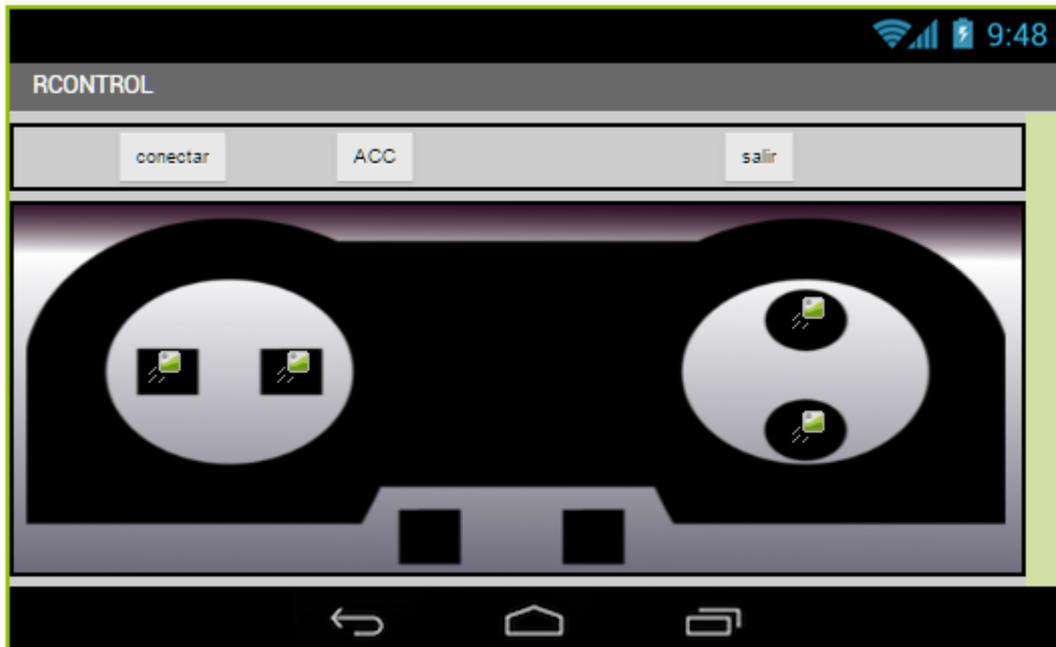


Figura 3. 24: Pantalla de mando RC de SUMOAPP.
Elaborado por: Autor

Posteriormente al crear las pantallas correspondientes a cada una de las opciones de nuestra aplicación se deberá programar con el diagrama de bloques para darle funciones específicas a todos los componentes que se encuentran en ella, como por ejemplo un botón, un gesto, etc.

En la figura 3.25 que se presenta a continuación se detalla el diagrama de bloques creado para la pantalla principal de nuestra app, correspondiente a una función sencilla, si se toca el botón AUTONOMO se apertura dicha pantalla, y la misma acción ocurrirá con el botón RC. En la figura 3.26, se presenta la programación en bloque de la pantalla autónomo, este enviará inalámbricamente un carácter por bluetooth, el modulo receptor se encargará de enviar vía serial al dispositivo Arduino para ser interpretado en el código de programación y designar la tarea correspondiente.

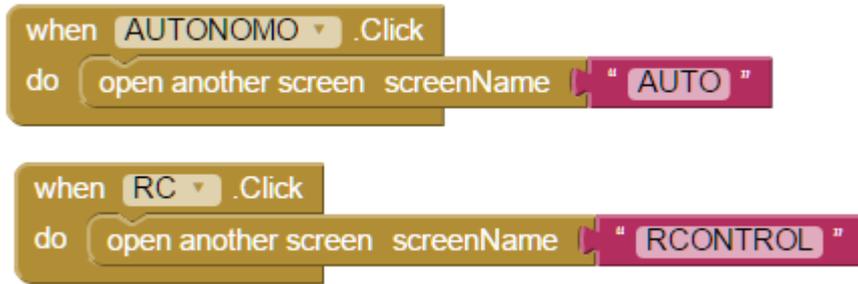


Figura 3. 25: Código de bloques de pantalla principal.
Elaborado por: Autores.

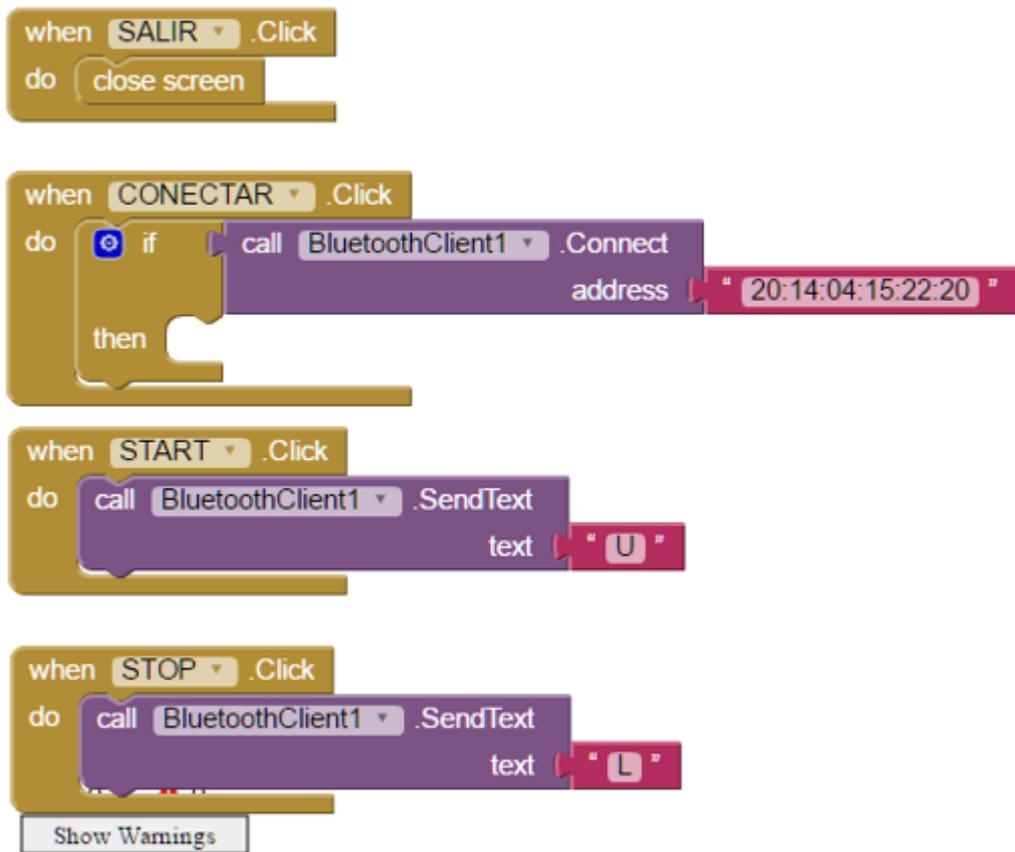


Figura 3. 26: Código de bloques de pantalla autónomo.
Elaborado por: Autor

Todas las funciones se ejecutarán de acuerdo al botón pulsado en el Smartphone o Tablet, así por ejemplo al seleccionar “CONECTAR” llamara al bluetooth cliente para conectar a la dirección mac correspondiente al módulo que se encuentra en la placa madre.

Los demás botones al ser seleccionado simplemente enviarán dicho carácter vía inalámbrica al módulo bluetooth. En la figura 3.27 se podrá apreciar la programación en bloque de la pantalla RC, es similar a la anterior, pero se adjunta nuevas características como cambiar de color al botón para confirmar conectividad, entre otras.

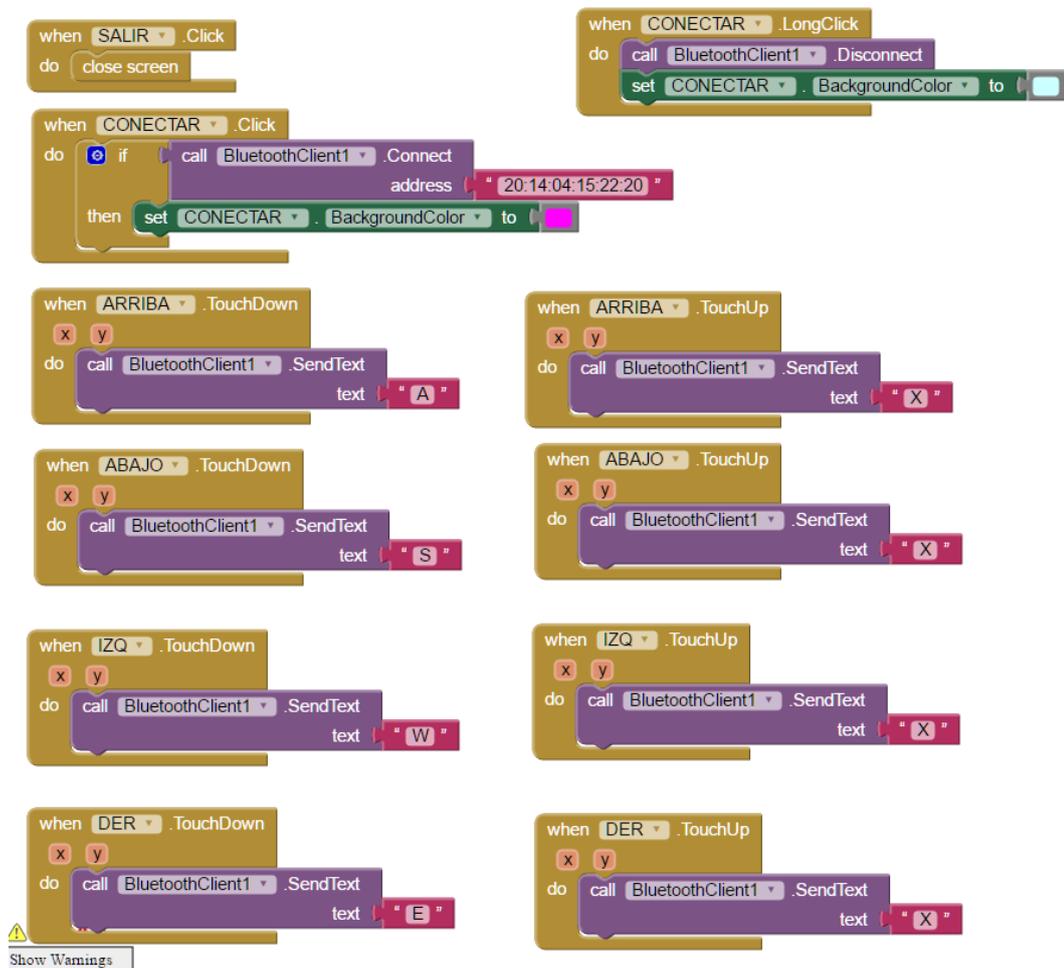


Figura 3. 27: Código de bloques de pantalla autónomo.
Elaborado por: Autor

Los botones conectar y salir cumplirán dicha función, en caso de mantener presionado el botón “CONECTAR” durante un tiempo prolongado lo

que se procederá a realizar es desconectar el enlace bluetooth que mantiene el Smartphone o Tablet al módulo.

En la pantalla de control RC no serán colocados botones para ejecutar los movimientos, simplemente se utilizará el touch que previamente se marcó en un área específica para determinar que en ese lugar se realizará una acción. Cada touchdown enviará un carácter específico que posteriormente en el código de programación de arduino se definirá el movimiento a realizar.

Cada touchup deberá enviar el mismo carácter en cualquiera de los movimientos realizados, con esto se especifica en el código que ha dejado de pulsar una orden y que es necesario parar las acciones que se encontraba realizando.

3.3.2 Código fuente de prototipo robot minisumo autónomo y RC

Para la programación del microcontrolador integrado en la placa embebida de Arduino se utilizará el sistema IDE de Arduino, ya que se encuentra integrada con varias librerías fáciles de usar y herramientas que nos ayudarán a la carga de nuestro código a su interfaz física. Iniciamos incluyendo las librerías y definiendo las variables a utilizar en nuestro sistema, figura 3.27.

```
MINISUMORCAUTO1 §
```

```
/* Software realizado para el club de robótica
 * Robofet en su interés de aplicar un prototipo
 * que funcione de dos maneras: Autónomo y Modo RC
 */
|
// DECLARACION DE LIBRERIAS A UTILIZAR
#include <Servo.h>
#include <SoftwareSerial.h>

// DECLARACION DE PINES, VARIABLES Y CONFIG DE LIBRERIAS
Servo motor1, motor2;
char dataIn;
int SENSOR= 16;
int x=0;
SoftwareSerial BT(9,8); // RX, TX recorder que se cruzan
```

Figura 3. 28: Cabecera de código fuente en Arduino
Elaborado por: Autor

Las librerías a utilizar son “Servo.h” y “SoftwareSerial.h”, ya que la primera nos ayudara a enviar las señales que permiten controlar los motores con el drive Sabertooth, el softwareserial nos ayuda a definir pines para su comunicación serial, Tx y Rx.

En la declaración de pines, variables y configuración de librerías se define servo motor 1 y 2 para enviar señales PWM al controlador de los motores, se declara una variable char dataIn que será la bandeja donde se recibirá los datos transmitidos por el Smartphone via bluetooth, se declara variables enteras int “SENSOR” para definir el pin donde se encontrará conectado el dispositivo de proximidad y “X” para el conteo de los cinco segundos reglamentarios en la categoría.

Y por último se declara los pines usados en la comunicación serial con SoftwareSerial, en este caso serán 9 y 8. A continuación, definiremos el void setup que no es más que la configuración inicial de todos los pines necesarios para la toma de variables y demás puertos como se muestra en la figura 3.29.

```
void setup() {  
  BT.begin(9600);  
  motor1.attach(5);  
  motor2.attach(6);  
  pinMode(SENSOR, INPUT);  
}
```

Figura 3. 29: Void setup de código fuente Arduino
Elaborado por: Autor

Lo que se define es la velocidad en baudios por el que trabajará el puerto serial, también los pines 5 y 6 serán por donde se enviará la señal PWM para controlar los motores y por último el modo del pin Sensor (16) que será de entrada.

```
// FUNCIONES DE MOVIMIENTO PARA EL PROTOTIPO  
void adelante() {  
  motor1.writeMicroseconds(2000); motor2.writeMicroseconds(1500);  
  delay(20);  
}  
void atras() {  
  motor1.writeMicroseconds(1000); motor2.writeMicroseconds(1500);  
  delay(20);  
}  
void derecha() {  
  motor1.writeMicroseconds(1500); motor2.writeMicroseconds(1000);  
  delay(20);  
}  
void izquierda() {  
  motor1.writeMicroseconds(1500); motor2.writeMicroseconds(2000);  
  delay(20);  
}  
void parar() {  
  motor1.writeMicroseconds(1500); motor2.writeMicroseconds(1500);  
  delay(20);  
}
```

Figura 3. 30: Funciones de movimiento.
Elaborado por: Autor

En la figura 3.30, se define las funciones para los movimientos que realizará a los motores gracias a la librería de servo previamente definida. Se coloca un retardo de 20 milisegundos después de ejecutar cualquier orden para estabilizar el puerto utilizado en nuestra placa Arduino.

```

// FUNCION DE DISPONIBILIDAD DE VARIABLES ENVIADAS POR BLUETOOTH
// SE EJECUTARAN LAS TAREAS DE MOVIMIENTO DEPENDIENDO DE CARACTER
// RECIBIDO POR EL BLUETOOTH
void COMBLUETOOTH() {
  if (BT.available())
  {
    dataIn = (char)BT.read();
    if (dataIn == 'W') {
      izquierda();
    }
    else if (dataIn == 'E') {
      derecha();
    }
    else if (dataIn == 'A') {
      adelante();
    }
    else if (dataIn == 'S') {
      atras();
    }
    else if (dataIn == 'X') {
      parar();
    }
    else if (dataIn == 'U') {
      //CONTEO DE CINCO SEGUNDOS
      //EN MODO AUTO
      while(x<5)
      {
        x++;
        delay(1000);
      }
    }
    else if(dataIn=='L'){
      // PARAR ROBOT EN MODO AUTO
      parar();
      delay(10000);
    }
  }
}

```

Figura 3. 31: Función que recibirá valores del módulo bluetooth y evaluará.
Elaborado por: Autor

Cuando las órdenes de la aplicación SUMOAPP son enviadas por bluetooth a nuestra placa Arduino es necesario relacionar cada carácter recibido a una función correspondiente sea el caso para que funcione de manera autónoma o controlada RC, para ello se necesita construir una función que permita preguntar qué valor tiene la bandeja de entrada como se muestra en la figura 3.31.

Al iniciar esta función, lo primero que se realiza es preguntar si está disponible el puerto serial del bluetooth, para ello lee el carácter enviado y procede a asignarle a la variable char previamente definida. Pasaremos a una etapa de pregunta que se evaluará la variable con el valor correspondiente para asignar una función en cada una.

En la sección que nos indica el conteo de 5 segundos es para realizar la tarea de inicio del robot autónomo al presionar el botón START en el Smartphone, al finalizar no realizará ninguna otra acción dentro de esta función, pero más adelante quedará definido que se desea que realice el robot.

Así mismo cuando se selecciona el botón STOP en la aplicación SUMOAPP se procede a detener el robot por 10 segundos, tiempo suficiente para poder retirar el prototipo del escenario de participación. A continuación, en la figura 3.31 se presenta el void loop que representa a la función principal de nuestro código fuente.

```

void loop() {
  COMBLUETOOTH();

  /* SE VERIFICA QUE SE ENCUENTRA EN MODO AUTO Y PROCEDE A REALIZAR TAREAS
  * DE BUSQUEDA Y ATAQUE
  */
  if(dataIn=='U'){
    if (digitalRead(SENSOR)==LOW) {
      derecha();
      delay(10);
    } else if(digitalRead(SENSOR)==HIGH) {
      adelante();
      delay(50);
    }
  }
}

```

Figura 3. 32: Función principal de código fuente.
Elaborado por: Autor

Esta función está en la obligación de iniciar inmediatamente COMBLUETOOTH (), ya que con ella se escaneará los valores recibidos por el puerto serial, al terminar dicha función se evalúa si la variable, dataIn posee el carácter correspondiente para efectuar el modo autónomo, en ser confirmada dicha acción procederá a realizar evaluación de sensores montados en la estructura para determinar movimientos de ataque y esquivo.

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.

4.1. Conclusiones.

- Con relación al diseño de la estructura del robot minisumo, la misma sufrieron alteraciones con respecto al peso, logrando así cumplir con las reglamentaciones de torneos nacionales e internacionales para minisumo de 500 g.
- El diseño electrónico realizado con el microcontrolador Atmega garantizó la comunicación por radio frecuencia e inalámbrica Bluetooth, y también garantiza el funcionamiento del robot en modo autónomo.
- Los dos robots fueron validados en competencias de robótica, la última realizada en la Escuela Politécnica Nacional, quedando en tercer lugar e incluye una acreditación para el torneo de UDENAR en Pasto, Colombia.

4.2. Recomendaciones.

- Implementar en las nuevas mallas de las Carreras de Telecomunicaciones y Electrónica la enseñanza de los microcontroladores Atmega utilizando plataformas de lenguaje de programación de alto nivel en código abierto (open source) como Arduino IDE.
- Implementar otros prototipos de robots móviles en base al diseño realizado, por ejemplo, un nuevo equipo de robot fútbol.

Bibliografía.

Arduino. (2016). ATmega 32U4-Arduino Pin Mapping. Recuperado el 15 de diciembre de 2017, a partir de <https://www.arduino.cc/en/Hacking/PinMapping32u4>

Arduino.cc. (s.f.). Arduino - Introduction., recuperado de: <https://www.arduino.cc/en/Guide/Introduction>

Atmel Corporation. (2016). ATmega164A DATASHEET COMPLETE., recuperado de: http://www.atmel.com/images/atmel-42712-atmega164a_datasheet.pdf

Atmel Corporation. (2012). 32-Bit Atmel AVR Microcontroller, recuperado de: <http://www.atmel.com/images/doc32058.pdf>

Atmel Corporation. (2015). ATMEL 8-BIT MICROCONTROLLER WITH 4/8/16/32KBYTES IN-SYSTEM PROGRAMMABLE FLASH DATASHEET., recuperado de: http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf

Atmel Corporation. (2013). 8-bit Atmel with 8KBytes InSystem Programmable Flash., recuperado de: http://www.atmel.com/Images/Atmel-2486-8-bit-AVR-microcontroller-ATmega8_L_datasheet.pdf

Dede, F. (2016). JSumo Sumo Robot Blog - Robot Sumo Projects, Parts. Recuperado a partir de <https://jsumo.com/>

FingerTech Robotics. (2016). Motors & Mounts - FingerTech "Silver Spark" 16mm Gearmotor. Recuperado el 12 de diciembre de 2016, a partir de <http://www.fingertechrobotics.com/proddetail.php?prod=ft-Sspark16>

Gualli Cujilema, Mayra Alexandra Véliz Intriago, Rocío Alexandra. (2016). Implementación de un Robot Mega Sumo Radio controlado y Autónomo utilizando microcontroladores PIC., recuperado de: <http://repositorio.ucsg.edu.ec/bitstream/3317/6401/1/T-UCSG-PRE-TEC-ITEL-144.pdf>

Guillem Arimany. (2011). Radio Frequency communication for Modular Robots., recuperado de: <https://upcommons.upc.edu/bitstream/handle/2099.1/14698/70000.pdf>

Joselyne Madeleine Beltrán Usho. (2016). evaluación de la precisión de sensores de corriente digitales basados en efecto hall, considerando los efectos de la temperatura e interferencias electromagnéticas, para aplicaciones de bajo coste de telemetría en la industria., recuperado de: <http://repositorio.ucsg.edu.ec/bitstream/3317/5482/1/T-UCSG-PRE-TEC-IECA-44.pdf>

Karla Erenia Jacome Guerrero. (2016). Diseño e implementacion de un robot movil soccer utilizando la tarjeta arduino nano y controlado mediante bluetooth., recuperado de: <http://repositorio.ucsg.edu.ec/bitstream/3317/5440/1/T-UCSG-PRE-TEC-ITEL-132.pdf>

Martin D. Stoehr. (2012). RF Basics., recuperado de: <http://pdfserv.maximintegrated.com/en/an/AN5300.pdf>

Milan Verle. (2008). PIC Microcontrollers., recuperado de: <http://ax7.com.br/wp-content/uploads/2014/07/PICMicrocontrollers.pdf>

Pololu. (2016). QTR-1RC Reflectance Sensor. Recuperado el 15 de diciembre de 2017, a partir de <https://www.pololu.com/product/959>

Steven F. Barrett. (2009). Embedded Systems Design with the Atmel AVR Microcontroller – Part I., recuperado de: <http://fit.mta.edu.vn/files/DanhSach/Embedded%20Systems%20Design%20-%20Part%201.pdf>

Tim Wilmshurst. (2007). Designing Embedded Systems with PIC Microcontrollers Principles and applications, recuperado de: <http://site.iugaza.edu.ps/engmoayed/files/2010/03/Designing.Embedded.Systems.with.PIC.Microcontrollers.eBook-S.pdf>



DECLARACIÓN Y AUTORIZACIÓN

Nosotros, **Robles Idrovo, Gabriel Augusto** con C.C: # 2300147556 y **Vaca Barberán, Gabriel Enrique** con C.C: # 0930646526 autores del Trabajo de Titulación: **Implementación de dos robots mini sumos utilizando sistemas de radio control “SABERTOOTH” y dispositivos de comunicación inalámbrica “BLUETOOTH”** previo a la obtención del título de **INGENIERO EN TELECOMUNICACIONES** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 13 de marzo de 2017

f. _____

Nombres: Robles Idrovo, Gabriel Augusto

C.C: 2300147556

f. _____

Nombres: Vaca Barberán Gabriel Enrique

C.C: 0930646526

REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	Implementación de dos robots mini sumos utilizando sistemas de radio control "SABERTOOTH" y dispositivos de comunicación inalámbrica "BLUETOOTH"		
AUTOR(ES)	ROBLES IDROVO, GABRIEL AUGUSTO; VACA BARBERÁN, GABRIEL ENRIQUE		
REVISOR(ES)/TUTOR(ES)	M. Sc. EDWIN F. PALACIOS MELÉNDEZ		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Educación Técnica para el Desarrollo		
CARRERA:	Ingeniería en Telecomunicaciones		
TÍTULO OBTENIDO:	Ingeniero en Telecomunicaciones		
FECHA DE PUBLICACIÓN:	13 de Marzo de 2017	No. DE PÁGINAS:	82
ÁREAS TEMÁTICAS:	Microcontroladores, Comunicaciones Inalámbricas		
PALABRAS CLAVES/ KEYWORDS:	MINISUMO, PIC, ATMEGA, ARDUINO, RADIOFRECUENCIA, BLUETOOTH		
RESUMEN/ABSTRACT (150-250 palabras):	<p>Podemos decir que nuestro trabajo está fundamentado en las transmisiones inalámbricas y la microelectrónica. Usamos microcontroladores los cuales están compuestos por dispositivos electrónicos que ayudan al funcionamiento del microcontrolador. Todo microcontrolador necesita un software que le ordene realizar una función específica. El enfrentamiento de robots minisumo es una batalla en la cual luchan 2 robots hasta sacar al oponente del dohyo, estos robots pueden ser autónomos y no autónomos los cuales pueden ser controlados por wifi, bluetooth y radiofrecuencia. La radiofrecuencia se la utiliza para el espectro electromagnético. Bluetooth es una comunicación inalámbrica de corto alcance. Mientras, que WI-FI es la tecnología más usada para la comunicación de dispositivos electrónicos de corto y largo alcance dependiendo su capacidad. Los dos prototipos implementados cumplen reglamentaciones para las participaciones en los próximos torneos de robótica. Finalmente, los diseños fueron validados en diferentes competencias logrando ubicarse entre los primeros lugares, y obteniendo certificaciones para torneos en Colombia y México.</p>		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: +593-9-990221764 +593-9-79108089	E-mail: gabriel_vaca92@hotmail.com gabriel_robles15@outlook.com	
CONTACTO CON LA INSTITUCIÓN:	Nombre: Palacios Meléndez Edwin Fernando		
COORDINADOR DEL PROCESO DE UTE	Teléfono: +593-9-68366762		
	E-mail: edwin.palacios@cu.ucsg.edu.ec		
SECCIÓN PARA USO DE BIBLIOTECA			
Nº. DE REGISTRO (en base a datos):			
Nº. DE CLASIFICACIÓN:			
DIRECCIÓN URL (tesis en la web):			